Automated Tracking Station

User's Manual

Automated Tracking Station User's Manual

Disclaimer

The controlled copy of this document is available on-line at http://www.wff.nasa.gov/~ats/index.html
Printed copies of this document are for reference purposes only. It is the user's responsibility to verify that the version of any printed documentation matches the on-line version.

This document was developed by the NASA/GSFC/WFF Real-Time Software Engineering Branch (Code 584). Any part of this document may be reproduced, stored in a retrieval system, and transmitted in any form and by any means, electronic or mechanical, including photocopying and recording, for any purpose without any permission, written or otherwise, of the Real-Time Software Engineering Branch.

This document was last modified on April 19, 2001. It corresponds to ATS release 3.5. Source code is archived on compact disc at NASA/GSFC/WFF. It is also managed under Microsoft Visual Source Safe administered by ViaSAT, Inc. personnel at Scientific-Atlanta facilities in Atlanta, Georgia.

Documentation Changes

April 9, 2002
  (1) Insert Note for operator intervention on schedule transfers p.18
  (2) Remove IP addresses
  (3) Modified ATS startup of 11 Meter Interface to every 30 seconds

February 1, 2002
  (1) Modified the Wotis PRF Format.
  (2) Corrected page number synchronization.

April 19, 2001
  (1) ATS web site URL is updated (above).
  (2) References to Scientific-Atlanta (SCC) software replaced by ViaSAT, Inc. (many pages.)
  (3) Added ATS **Help** personnel contacts (below).
  (4) Added GRM start-up procedures.
  (5) Added GUI information under Monitor and Control.
  (6) References to Device Modes added to Device Interfaces.
  (7) Added GUI information to MSC10693 Analog Switch referencing Load New button and
       functionality.
  (8) Added GUI information to GDP911 Digital Switch referencing Load New button and
        functionality.
  (9) Added reference to Access Modes under Chapter 3.
  (10) Added title bar description.
  (11) Added information as to how operations accesses the profile editor.
  (12) Added Load New button under Device Specific Information.

January 7, 2001
  (1) ATS web site URL is updated (above).
  (2) References to Scientific-Atlanta (SCC) software replaced by ViaSAT, Inc. (many pages.)
  (3) Added ATS **Help** personnel contacts (below).

January 24, 2000

August 9, 1999
          Modified Appendix J to explain ATS zip file installation procedure.

August 2, 1999
  (1) Added information regarding SCC/ATS shutdown procedure (pp. 10,15).
  (2) Added information regarding operational profile re-naming convention (p. 13)
  (3) Added information for the duplicate tape shipping report file (p. 20).
  (4) Added Appendix K: Duplicate Tape Report Format (p. 17).

July 12, 1999
  (1) Added MSC-10693 and GDP-911 switch device documentation (pp 23-24).
  (2) Removed specific IP addresses in Station Status Broadcaster (p. 95).
  (3) Added Appendix I: Master/Node System Administration
  (4) Added Appendix J: ATS Software Installation Guide

Help

You can contact software development via e-mail at the following addresses:
Edward.Payne@csoconline.com   Jeanette.Smolinski@csoconline.com
Lakeesha.Ruffin@csoconline.com   Melvin.Ritter@csoconline.com
Jeffrey.L.Dorman.1@gsfc.nasa.gov DeAndre.Hayward@csoconline.com

Table of Contents

List of Figures

Chapter 1 : Introduction

In 1993, the decision was made to begin upgrading and automating the Wallops Orbital Tracking Station (WOTS). The objective of the upgrade is "…to automate the existing… station… to reduce operating costs and improve availability." Specific requirements included providing centralized control and monitoring of WOTS resources, providing automated control and monitoring of the station, integrating existing resources into the automated station, and insuring the maintainability of the developed system.

As the effort to automate WOTS progressed, the applicability of the developing system at other tracking stations was recognized. Therefore, the automation objective was expanded to include stations at McMurdo, Antarctica (MGS), Svalbard, Norway (SGS) and Poker Flat, Alaska (AGS). Furthermore, it was determined that minimizing the differences between the various systems would help maximize their maintainability.

What has evolved is the development of a distributed, generalized system that employs customized software and Commercial Off-the-Shelf software (COTS) to accomplish the automation objective. The system is comprised of a network of Master stations, Remote Node stations, and integrated independent systems.

What is this manual?

This manual is intended to be a useful reference to both the new and the experienced Automated Orbital Tracking Station user. It includes a brief overview of the automation software, detailed instructions for using the software, and tips for troubleshooting problems that may occur. It is assumed that the user of this manual (and, presumably, the user of the automated tracking station) has some experience with Windows based graphical user interfaces.

What is the Master?

The Master station is the seat at which centralized control and monitoring of the station is exercised. The Master is a collection of integrated processes that coordinate communication between the operator, the Remote Nodes, and the integrated independent systems in order to do the following:
- accept, ingest, and interpret an operational schedule
- create and modify support profiles
- distribute configuration information to tracking station resources
- manage setup, start, stop, and takedown of support activities
- monitor support progress
- provide an interface to all automated station resources
- generate post-support reports
- monitor and report station status to remote users

An automated tracking station is expected to have multiple, coequal and redundant Master stations.

What is a Remote Node?

The Remote Node computers exercise low-level control of a subset of tracking station resources, taking direction from and reporting to the Master as appropriate to the resources and their status. The equipment connected to and controlled by a Remote Node varies from tracking station to tracking station. Equipment types typically controlled by a Remote Node include bit synchronizers, frame synchronizers, filters, simulators, receivers, combiners, matrix switches, modulators, and demodulators. Furthermore, there is no limitation imposed on the distribution of equipment across the Remote Nodes at a tracking station by the automation software.

The Remote Nodes include a complement of processes that coordinate communication between the Master, the equipment, and, in some cases, a user in order to do the following:

- ✓ Report on equipment availability
- ✓ accept equipment configuration information from the Master
- ✓ handle request, setup, start, stop, and takedown commands from the Master
- ✓ report equipment status
- ✓ log and distribute equipment support information
- ✓ provide local control and monitoring of equipment in the event of a Master-Node network failure

An automated tracking station is expected to have one or more Remote Nodes.

What is an integrated independent system?

When possible, the automation effort has included integrating existing systems and/or commercial off the shelf software (COTS) into the new system; currently the systems are being sustained using Visual C++ 5.0 concurrently in versions 3.4 and 3.5. In integrating independent systems, a minimal set of interface requirements were levied against the original developer or vendor in order to provide a scheme by which the Master can issue commands and request status. Examples of integrated independent systems include the Scientific Atlanta 11-meter antenna station control computer (SCC), Avtec 1000 Programmable Telemetry Processor, and TSI data strippers.

The Automation Cycle

The automation required to accomplish the objectives during a scheduled support follows a fixed sequence of events managed by the Automated Tracking Station (ATS) software processes which contains the Acquisition of Signal (AOS) and Loss of Signal (LOS). This cycle includes:

Initialization (AOS – ~15 minutes)
(1) Master PC configures the Station Control Computer (SCC), sometimes referred to as the *11meter node*.
   (a) Create, deliver and load a specially formatted schedule file containing AOS, LOS, configuration identification number and Sony/Ampex recorder start/stop times. This file is typically delivered via FTP to the directory */users/aaas/etc/remote*. This file path, as well as the *11meter node* FTP profile is identified in the *StationAssets.dat* text file. (See *Starting the Station Assets Editor* in **Chapter 4 – Configuring the Master for a Station**.).
   (b) Create, deliver and load a specially formatted file containing ephemeris vector information. This file is typically delivered to the directory */users/aaas/etc/remote*. This file path, as well as the *11meter node* FTP profile is identified in the *StationAssets.dat* text file. (See *Starting the Station Assets Editor* in **Chapter 4 – Configuring the Master for a Station**.).
(2) Master PC configures the Receiver node and Data Stripper Controller node computers. A specially formatted file is delivered to these nodes that contain scheduler and configuration file information. These files are delivered to file paths and FTP profiles identified in the *StationAssets.dat* text file. (See *Starting the Station Assets Editor* in **Chapter 4 – Configuring the Master for a Station**.).
(3) Master PC requests resources from a remote node computer for an upcoming support. These resources could be bit synchronizers, Metrum recorders or any other device connected to a remote node. Resource assignment must be granted in order for subsequent phases in the automation cycle to succeed. These resources are dedicated until released from automation during the takedown stage.
(4) Master copies device-specific configuration files to the remote nodes. These files contain device-specific parameters – like support ID, receive pad ID and doppler mode for the Tracking Data Formatter.
(5) ATS initialization time is identified by the operator using the *ProfileEditor* utility (see **Chapter 5 Profiles**). The minimum initialization time allowed is 150 seconds prior to scheduled AOS to accommodate SCC antenna requirements. The maximum time allowed is operations dependent.
(6) TCP/IP socket communications are initiated from the Master PC to remote status clients. These clients are identified by IP address prescribed by the operator using the *ProfileEditor* utility. Successful socket connections are followed immediately by a transmission of station status. (See **Appendix F: Station Status Broadcasts**).

Setup (AOS - ?? minutes)

Master sends commands to remote nodes to load configuration files copied during the initialization phase. The setup command is delivered immediately upon resource assignment acknowledgement.

Start (AOS - ~60 seconds)
Master sends commands to the remote nodes to activate, enable and start each unit of equipment. Equipment must be in a setup state. The equipment start time is identified by the operator using *the ProfileEditor* utility (see **Chapter 5 Profiles**).

Stop (LOS + ~5 seconds)
Master sends commands to the remote nodes to disable and stop each unit of equipment. Equipment stop time is identified by the operator using *the ProfileEditor* utility (see **Chapter 5 - Profiles**).

Takedown (LOS + ?? minutes)
The equipment takedown state is the final stage of automation. This stage is started immediately after the equipment stop state is acknowledged.
(1) Nodes close all device status logs created during automation and release equipment from assignment.
(2) Nodes copy all device status logs back to the *c:\temp* folder on the Master. This includes *11meter* passlog, binary tapelog (if Sony/Ampex recorders were required) and message files.
(3) The node controlling the Avtec PTP device creates the meta-files and forwards them on to the local Standard Autonomous File Server (SAFS) identified in the *lmhosts* table.
(4) The Avtec PTP device is loaded with a *nothing.dtp* desktop file in order to clear all stream information. (See **Appendix E: PTP Automation**).

Pass Results (LOS + ?? minutes)
(1) Tracking Data Formatter, Metrum recorder, PTP, SCC and Data Stripper Controller node status logs are processed to create a pass results file. This file is copied back to the WOTIS scheduling office (see **Appendix C: Pass Results Format**).
(2) Universal tracking data file (UTDF) information is copied via FTP to remote clients.
(3) 30-day old logs, schedules and other text files are automatically deleted.

Chapter 2 : System Overview

The Master includes a collection of Year-2000 compliant independent processes that cooperate to provide centralized control and monitoring of the tracking station. The system includes the *core processes*, *utilities*, and *device interfaces*. The core processes include those processes that must be running on at least one Master in order for the automated tracking station to accomplish the operational schedule. Utilities include station, support, and device configuration editors with which station specific characteristics can be identified and anticipated supports can be defined. Device interface applications provide an interface with which the user can exercise control, monitor the status, and define the configuration of a device. They can serve as both pre-support utilities and support critical processes.

ATS Core Processes

Some of the Master core processes include a graphical user interface (GUI); others are transparent to the user.   The Master includes the following core processes:

- Monitor and Control (*c:\Master\MonitorAndControl.exe*)
- WOTIS Interface (*c:\Master\WOTISInterface.exe*)
- Scheduler (*c:\Master\Scheduler.exe*)
- General Resources Manager (*c:\Master\Grm.exe*)
- 11-meter Station Control Computer (SCC) Antenna Interface (*c:\Master\11mInterface.exe*)
- Receiver node Interface (*c:\Master\RNInterface.exe*)
- Data Stripper Controller Interface (*c:\Master\DSCInterface.exe*)
- Pass Results Compiler (*c:\Master\PassResultsCompiler.exe*)
- SAFS Heartbeat (*c:\Master\ SAFSHeartbeat.exe*)

ATS Startup/Shutdown Procedures

The following procedure must be followed when stopping ATS and the SCC processes:
(1) Stop the ATS applications on the Master PC by selecting *Shutdown Master Subsystem* from the System menu.   This action will close all network connections to the SCC and node computers and stop all ATS processes executing on the Master.
(2) Stop the General Resource Manager (GRMs) applications on the node PCs**. If re-connection between the Master and the SCC is sought, the GRMs do not have to be stopped.**
(3) Stop the SCC processes by typing "Stop" at a user prompt. SCC problems should be documented and sent ViaSAT personnel by first saving *home/aaas/etc/message.file* ("*cp /home/aaas/etc/message.file /home/aaas/etc/message-save.file*");

The following procedure must be followed when starting or re-starting ATS and the SCC processes:
(1) Verify that all systems are idle awaiting a Start command.
(2) Start the SCC processes by typing "Start" at a user prompt. SCC processes will start and a GUI will display. Select *login* from the Session menu.
(3) Start the General Resource Applications (GRMs) on the ATS node computers by first initializing the devices GRM and second the operations GRM (see GRM discussion below on page 14.)
(4) Finally, when all GRM applications are started successfully, start ATS on the Master computer by logging in as user "Operator" or by simply selecting the "*MonitorAndControl*" icon in the desktop folder. Select "Yes" for the question "Make this the Scheduling Master ?" in order to start connections with the SCC.

A brief description of each of the core processes is provided here; specific instructions for using each of the processes can be found in later chapters of this manual.

Monitor and Control *(c:\Master\MonitorAndControl.exe)*

MonitorAndControl.exe (M&C) starts ATS core processes automatically and coordinates communication between the Master, user, Remote Nodes, and any integrated independent systems. *MonitorAndControl* starts the following master ATS core processes automatically:

(1) *c:\Master\WOTISInterface.exe*
(2) *c:\Master\Scheduler.exe*
(3) *c:\Master\Grm.exe*
(4) *c:\Master\SAFSHeartbeat.exe*

Successful startup of *MonitorAndControl* and related core processes is displayed in the *MessageLog* window as:

(04/06/1999 (096) 14:39:20) Monitor and Control started.
(04/06/1999 (096) 14:39:20) Operator is the current user.
(04/06/1999 (096) 14:39:20) The Scheduler process has been started.
(04/06/1999 (096) 14:39:20) The WOTIS Interface process has been started.
(04/06/1999 (096) 14:39:20) The General Resources Manager has been started.
(04/06/1999 (096) 14:39:20) The SAFS Interface has been started.
(04/06/1999 (096) 14:39:20) The Scheduler process is active.
(04/06/1999 (096) 14:39:20) The WOTIS Interface process is active.
(04/06/1999 (096) 14:39:20) The General Resources Manager is active.

The *MonitorAndControl* process is the central operation of ATS. It is responsible for the following actions:

(1) **Communication with ATS processes executing on node computers**. This communication involves the exchange of messages between the Master and node computers through the General Resource Manager (GRMs) applications. These message exchanges are conducted over Window NT named pipes using a RS-232 port and are critical to device status collection and their transmission to remote clients.

(2) **Monitoring of events related to the Automation Cycle**. The ATS design currently manages the initialization, setup, start, stop and takedown of one support at a time as requested by the ingested schedule of events.   Status regarding the automation cycle and communication with the remote node computers are displayed in the *M&C* message log window and archived for thirty days in time-tagged files at *c:\Master\MasterLogs*.

(3) **Preparation of the SCC schedule and ephemeris files**. Initialization of supports that require the SCC forces *MonitorAndControl* to prepare two files for delivery to the SCC via FTP. The remote schedule file (*c:\Master\Schedule\SciAt11mPass.scd*) contains AOS, LOS, and configuration ID number and recorder information. The AOS and LOS times are extracted from the ATS schedule and placed in, *c:\Master\Schedule\AutomatedOps.scd*. The SCC recorder units and configuration number are extracted from the *ReceiveAntenna.rcv* file editable in the support profile. The remote ephemeris file (*c:\Master\Schedule\SciAt11mEph.scd*) contains the browser, true-of-date or IIRV pointing vector extracted from the *c:\Master\Schedule\AutomatedOps.scd* file. These remote files are sent to the SCC using FTP profiles identified in the *c:\Master\Station\StationAssets.dat* setup file.

(4) **Preparation of the Data Stripper Controller (DSC) schedule file**. Initialization of supports that require the TSI Data Stripper forces *MonitorAndControl* to prepare one file for delivery to the DSC node PC via FTP. The remote schedule file (*c:\Master\Schedule\DSCnode.scd*) contains AOS, LOS and configuration name. The AOS and LOS times are extracted from the ATS schedule, *c:\Master\Schedule\AutomatedOps.scd*. The configuration name is identified using the ProfileEditor application. These remote files are sent to the DSC node using FTP profiles identified in the *c:\Master\Station\StationAssets.dat* setup file.

(5) **Coordination of user actions through a graphical user interface (GUI).** The GUI allows the operator the capability to stop and re-start individual ATS processes, insert and delete scheduled events and edit base and support profiles without terminating ATS.

(6) **Transmission of ground station equipment status to remote clients**. At support initialization, M&C automatically starts a station status transmission thread and begins sending the packets to remote

clients. The clients are identified by IP address and port number using the *ProfileEditor* utility.   The status data are transmitted one packet per second as a TCP/IP socket message. Also, it is determined that status is known by the viewing of the active pendulum swing on the status view window. Clients receiving station status data should use the *StationStatusDisplay* utility, described later in this chapter, to view the data. Status packet contents and formats are explained in **Appendix F: Station Status Broadcasts**. Execution of the station status thread dedicates the Master as the "*ReportingMaster*". Status transmission is stopped at the ATS takedown state.

Successful startup of station status is displayed in the *MessageLog* window as:

(03/23/1999 (082) 23:32:45) The Station Status Broadcaster has been started.

The *Initialization* phase for station status transmission is displayed in the *MessageLog* window as:

(03/24/1999 (083) 13:53:47) Starting station status monitor thread.

Any failure to connect to remote clients is displayed in the *MessageLog* window as:

(03/23/1999 (082) 23:32:49) Failure to create a socket on 128.144.56.78, port #205 testing.

WOTIS Interface *(WOTISInterface.exe)*

The WOTIS Interface manages all communication between the Wallops Orbital Tracking Information System (WOTIS) and the Master. WOTIS is the main scheduling partner for a Master. *WOTISInterface* is started automatically by *M&C* and idles forever as it anticipates a socket message from WOTIS notifying the Master that a new operational schedule has been delivered via FTP.   A valid account for user *wotrspw* must be installed on the Master for successful FTP. Successful startup of *WOTISInterface.exe* is displayed in the MessageLog window as:

(04/06/1999 (096) 14:39:20) The WOTIS Interface process has been started.
(04/06/1999 (096) 14:39:20) The WOTIS Interface process is active.

 The specific name and contents of the schedule delivered by the scheduling office, the specific contents of the notification message, and the socket number to which the notification message should be delivered are defined in **Appendix B: WOTIS – Master Communication**.

The following sequence of events occurs when *WOTISInterface* receives notification of delivery of a new schedule file, *c:\Master\Schedule\wotrs.mas* via FTP:

(1) A copy of *wotrs.mas* is created at *c:\Master\Schedule\Archive*. The file copy is named according to the time of archive using the following convention, *WOTRS_yyyymmdd_hhmmss.scd*. This file copy is beneficial for trouble-shooting ATS/Scheduling problems and is maintained for thirty days.
(2) The *wotrs.mas* contains requests for a variety of antenna supports. It is processed and separated into daily schedule files at c:\Master\Schedule. These files are named using the convention, *WOTRS_yyyymmdd.scd* .   ATS removes all of these daily schedule files before creating new ones. Thus, there is no limitation to the number of days included in the schedule nor is there any requirement that schedule files be delivered in chronological order. In other words, the scheduling office can send updates or corrections for a specific day without interfering with the schedule for an unaffected day.
(3) The daily schedule files are processed and combined into one station operational schedule, or *c:\Master\StationOps.scd* . These files, as well as the wotrs.mas, are re-written on the next schedule delivery.
(4) The *WOTISInterface* sets an event flag that notifies another Master core process, *Scheduler.exe*, that WOTIS schedule processing has been completed.

(5) *WOTISInterface* returns to an idle state and awaits delivery and notification of the next schedule.

Scheduler *(c:\Master\Scheduler.exe)*

Scheduler is a core process started automatically by *M&C*. Generally, it is responsible for WOTIS schedule processing. No conflict resolution is applied during schedule processing. WOTIS Scheduling engineers must guard against overlapping supports. A 15-minute gap between consecutive supports is recommended for reliable ATS behavior. Successful startup of *Scheduler* is displayed in the *MessageLog* window as:

(04/06/1999 (096) 14:39:20) The Scheduler process has been started.
(04/06/1999 (096) 14:39:20) The Scheduler process is active.

Scheduler idles forever and waits for one of the following conditions to occur:

(1) *WOTISInterface* notifies it that a new WOTIS schedule has been delivered.
(2) Operator uses the utility *c:\Master\ManualNotification.exe* to manually process schedule files.
(3) Operator inserts a new schedule support using the utility.
(4) Operator deletes a scheduled support.
(5) Operator modifies the start and stop event times for a support.
(6) Operator modifies the contents of an operational, or support, profile.
(7) The time is reached for an automation cycle-related event (*Initialization*, *Start*, *Stop* or *PassResults*). The idle time to wait for the instance of one of these events is one second.

ATS prohibits schedule ingestion during an on-going support (between *Initialization* and *Takedown*). Instead, schedule processing is delayed until ten seconds after the *PassResultsCompiler* process terminates. Ingestion of new or modified supports causes all schedule-related files to be processed.   Processing of schedule files is displayed in the MessageLog window as:

(04/06/1999 (096) 14:39:20) Ingesting WOTIS schedule and user-insertions/deletions.

 The set of files processed during schedule ingestion includes:
(1) *c:\Master\Schedule\StationOps.scd*
(2) *c:\Master\Schedule\UserModifications\INSERT…*
(3) *c:\Master\Schedule\UserModifications\DELETE..*
(4) *c:\Master\Schedule\UserModfications\MODIFY…*

Schedule ingestion in *Scheduler* is accompanied by several activities. These include:
(1) All schedule-related files are processed in order to create the operational schedule file, *c:\Master\Schedule\AutomatedOps.scd* . *Scheduler* filters out all events not supported by ATS when creating this file. This file contains all activities that ATS-supported antennas can manage at the particular ground station.   It is over-written and updated on the next schedule delivery, as well as at post-pass, after the conclusion of each support.
Operational, or support, profiles are created that match information requested in the operational schedule file. These support profiles contain information regarding types of resource devices (bit synchronizers, programmable telemetry processors, etc.) that must be setup by ATS to accommodate a support. The profiles are created automatically at *c:\Master\OpProfiles* using a reference list of base profiles at *c:\Master\Profiles* created by operations personnel.   The base profile used to create the support profile for any given event is derived by a simple matching algorithm that compares the contents of the operational schedule to similar information contained in the base profile. See **Appendix B: WOTIS – Master Communication** for an explanation of each schedule parameter. The matching criteria include:
    (a) WOTIS-recognized satelliteID,

(b) receive or transmit antenna designator, and

(c) TR code.

Operational profiles are created automatically and named according to the following convention:
SAT_MO_DD_YYYY_HH_MM_SS_AN_TR,

where	SAT	= WOTIS-recognized satelliteID (see **Appendix B – WOTIS…**);
	MO	= scheduled 2-digit AOS month
	DD	= scheduled 2-digit AOS day-of-month
	YYYY	= scheduled 4-digit AOS year
	HH	= scheduled 2-digit AOS hour
	MM	= scheduled 2-digit AOS minute
	SS	= scheduled 2-digit AOS second
	AN	= scheduled 2-digit receive antenna designator.
	TR	= TR support code.

Operational profiles are *not* re-created upon schedule delivery and ingestion. They are removed three days after their scheduled AOS.

(2) A chronological list of ATS-supported events is created at c:*Master\Schedule\ThingsToDo.txt* . This text file is over-written and updated on the next schedule delivery, as well as at post-pass, after the conclusion of each support. The following is a sample list of events from the text file:

(364) 12/29/2000 at 20:35:49 GMT - Initialize-Setup for QST_12_29_2000_20_38_29_00_TR1 (1-related events).

(364) 12/29/2000 at 20:37:34 GMT - Start BVLDS for QST_12_29_2000_20_38_29_00_TR1 (event #1).

(364) 12/29/2000 at 20:37:44 GMT - Start BitSynchronizer for QST_12_29_2000_20_38_29_00_TR1 (event #1).

(364) 12/29/2000 at 20:37:46 GMT - Start MatrixMSC10693 for QST_12_29_2000_20_38_29_00_TR1 (event #1).

(364) 12/29/2000 at 20:37:57 GMT - Start GDP911 for QST_12_29_2000_20_38_29_00_TR1 (event #1).

(364) 12/29/2000 at 20:38:18 GMT - Start ApogeeTDF for QST_12_29_2000_20_38_29_00_TR1 (event #1).

(364) 12/29/2000 at 20:40:29 GMT - Stop-Takedown GDP911 for QST_12_29_2000_20_38_29_00_TR1 (event #1).

(364) 12/29/2000 at 20:40:29 GMT - Stop-Takedown BVLDS for QST_12_29_2000_20_38_29_00_TR1 (event #1).

(364) 12/29/2000 at 20:40:29 GMT - Stop-Takedown BitSynchronizer for QST_12_29_2000_20_38_29_00_TR1 (event #1).

(364) 12/29/2000 at 20:40:29 GMT - Stop-Takedown MatrixMSC10693 for QST_12_29_2000_20_38_29_00_TR1 (event #1).

(364) 12/29/2000 at 20:40:29 GMT - Stop-Takedown ApogeeTDF for QST_12_29_2000_20_38_29_00_TR1 (event #1).

(364) 12/29/2000 at 20:40:39 GMT - PassResults for QST_12_29_2000_20_38_29_00_TR1.

(3) The *M&C* process is notified in order to update the operational schedule window display.

General Resources Manager *(c:\Master\Grm.exe, c:\Wff\Grm.exe)*

The General Resource Manager (*GRM, "the Grim"*) allows a resource to be accessed and shared by multiple users across a network. A resource may be a hardware device attached to the computer, such as a Metrum BVLDS recorder, or a software module, such as the support operations for each device.

Each node must be executing two unique *GRMs* . One *GRM*, the *Devices GRM* (*c:\Wff\Grm.exe*), establishes and manages connections with all ground station equipment. A second *node GRM* (also at *c:\Wff\Grm.exe,* but with different command line arguments), the *Operations GRM,* is responsible for exchanging device-specific messages with the equipment. The communication protocol is either RS-232 serial or TCP/IP network messages.

The ground station master must always execute a *GRM* as an ATS core process. This *Master GRM (c:\Master\Grm.exe)* is started automatically by the *M&C* process and exchanges automation cycle messages with the node *Operations GRM* using Windows NT message pipes. Typical *Master GRM* messages may be "Load configuration file xxx on TDF unit 1…" or "Start PTP unit 2…". The *Master GRM* must be executing in order to qualify the machine as the *Scheduling Master*. The *Master GRM* expects to receive a response from the *node Operations GRM* for every command request.

The following general procedures outline the method when starting and stopping the *GRMs*:
(1) Node-controlled equipment must be in an idle and remote-mode state and waiting for a GRM connection on startup.
(2) All *node GRMs* must be started successfully before the *Master GRM*.   A successful *node GRM* startup is displayed as "Creating the shutdown event. Ready ---------------------------".
(3) The *Devices GRM* is started automatically upon node user *Operator* login. Any failure to communicate with equipment on startup of the Devices GRM is displayed in the *GRM* window. For instance, failure to establish connections with a frame synchronizer model GDP225D unit 2 is shown as "Unable to reset resource \GDP225D\2."
(4) The *Operations GRM* must be started manually.
(5) On nodes, the *Devices GRM* must always be started before the *Operations GRM* .
(6) The *Master GRM*, as well as other ATS core processes, are started automatically at Master startup. Successful startup of the *Master GRM* is displayed in the *MessageLog* window as:

 (04/06/1999 (096) 14:39:20) The General Resources Manager has been started.
 (04/06/1999 (096) 14:39:20) The General Resources Manager is active.

(7) *Node GRMs* are stand-alone; they **do not** require execution of the *Master GRM* to control equipment. The *Master GRM* is required only for support scheduled through ATS on the master.
(8) *Master GRMs* must be stopped before *node GRMs*.
(9) Failure to successfully start a *node GRM* will cause the *Master GRM* to fail. This failure causes ATS to attempt multiple re-startups of another Master GRM, usually producing a fatal ATS error on the Master.

The *Master GRM* is directed to connect to nodes that are named in a registry file located at *c:\Master\Grm.reg*. *Node GRMs*, however, need only a clue regarding the set of connected equipment in order to begin polling status and sending commands. This clue is contained in two registry files located at *c:\Node\Regs\DevicesGrm.reg* and *c:\Node\Regs\OperationsGrm.reg*. See **Appendix H: Configuring the GRM Registry** for more information regarding *GRM* setup.

Station Control Computer (SCC) Interface *(c:\Master\11mInterface.exe)*

The 11-meter Antenna Interface is a multi-threaded Windows NT application that coordinates communication between the Master and the Scientific Atlanta (SA) 11-meter Antenna control computer. This system is typically referred to as the Station Control Computer (SCC). This process is responsible for forwarding and loading schedule and ephemeris requirements to the antenna system.

Communication between the Master PC and SCC is accomplished using one 2-way TCP/IP socket connection   Successful socket communication requires the following Windows NT *hosts* and *services* configuration file changes:

(1*)* Socket communication service identification in *c:\winnt\system32\drivers\etc\services.*

```
nasa_recv      3011/tcp    # 11meter NASA message receive service
nasa_send      3012/tcp    # 11meter NASA message send service
```

(2) Socket communication IP address, hostname and aliases in *c:\winnt\system32\drivers\etc\hosts.*

```
150.xxx.xxx.xx  wgs11m-1                master_computer
150.xxx.xxx.xx  wgsmaster1              remote_host    thisone
```

The alias, *master_computer,* is a required alias for the SCC.   The alias, *remote_host*, is a required alias for the ATS Master PC.   These two alias names are expected in order for the ATS *11mInterface* application, and SCC process, *rci_remote*, to cooperate. The alias *thisone* is used for Data Stripper Controller interface execution.

Successful delivery of schedule and ephemeris files depends on the existence of WS_FTP32 profiles identified by the operator in the *c:\Master\Station\StationAssets.dat file*.   See **Chapter 4 – Configuring the Master for a Station. Starting the Station Assets Editor**.

The socket is initialized in order to send messages from the Master to the SCC.   These messages include:

(1)   A request for socket connection.
(2)   A request for remote login using SCC account *remotepc.*
(3)   A request for directory names containing 11meter passlogs, tapelogs and configurations.   These directory names are replaced by Station Assets information (see **Chapter 4 – Configuring a Master for a Station**).
(4)   A request for directory names where remote schedules and ephemeris files should be delivered. These directory names are replaced by Station Assets information (see **Chapter 4 – Configuring a Master for a Station**).

The above four messages are delivered when the *11mInterface* process begins execution. The following three messages are delivered at support *Initialization* time (usually AOS – 10 minutes) and after preparation and FTP of the remote schedule (*c:\Master\Schedule\SciAt11mPass.scd*) and ephemeris (*c:\Master\Schedule\SciAt11mEph.scd*) files by the *MonitorAndControl* process:

(1)   request for *control upgrade* status in order to prepare for loading schedule and ephemeris files;
(2)   request to *load* the remote ephemeris, then schedule file.

Note: Operator is to wait 30 seconds before intervening for schedule transfers.

(3)   request for *control downgrade* status in order to release remote control back to the SCC.

The TCP/IP socket connection is also responsible for listening and receiving instrument status packets, as well as granting approval for *login* and *control upgrade* requests. The initialization of this socket

constitutes successful application startup and forces the Master PC to function as the "*Scheduling Master*". See **Chapter 6: Monitoring and Controlling – What is the Scheduling Master ?** for more information regarding scheduling functions.

Information from SCC recorder status packets is saved in the folder *c:\Master\11MeterTapes* in files named *S-A-Recorderx.txt* (where 'x' represents unit number). Recorder status file updates are determined by the parameter *Status-Update-Rate-secs* specified in *c:\Master\11MeterTapes\RecorderProperties.txt*; 120 seconds is the default rate.

This process is started when the user dedicates the Master as the "*Scheduling Master*" by answering yes to that option. Successful startup of this process requires prior startup and execution of the Scientific-Atlanta UNIX processes on the SCC and is displayed as an aqua-colored Master icon on the block diagram and marked in the *MessageLog* window as:

(04/06/1999 (096) 14:39:20) The 11m Antenna Interface process has been started.
(04/06/1999 (096) 14:39:20) This Master is now the Scheduling Master.
(04/06/1999 (096) 14:39:20) 11 m Interface... Send socket initialized for AGS11m_Host at AGS .
(04/06/1999 (096) 14:39:28) 11 m Interface... Listen socket initialized for AGS11m_Host at AGS .
(04/06/1999 (096) 14:39:33) 11 m Interface... Sending login request to S-A SCC XIFSOCK.
(04/06/1999 (096) 14:39:33) 11 m Interface... Remote login to 11m system approved.

Termination of the SCC processes during execution of the *11mInterface.exe* on the Master will force automatic shutdown of *11mInterface.exe*. ATS will attempt to restart *11mInterface.exe* every 30 seconds when a forced shutdown occurs. ATS design considers this procedure an abnormal one because the SCC is regarded as a node computer. Node computers must be stopped after ATS processes have shutdown on the Master computer. An abnormal interruption of the TCP/IP sockets connections between ATS and the SCC should be resolved by the following procedure:

(1) Stopping the ATS applications on the Master PC;
(2) Stopping the SCC processes. (If SCC problems are to be documented, copy the file */home/aaas/etc/message.file* to */home/aaas/etc/message-save.file* and send to ViaSAT personnel immediately.) A subsequent *Start* of the SCC will remove and re-create the *message.file*.
(3) Log out of the Master PC;
(4) Start the SCC;
(5) Login to the Master PC and execute ATS by starting the *MonitorAndControl* process.

11mInterface can also be started as an offline-ATS utility with status updated in a GUI by executing the process with the command line argument "ShowNow".

Interaction with the SCC from the Master can also be achieved by executing *c:\Master\SA11Meter#1.xs*. This *Xsession* script requires installation of Exceed 5.0 and enables an X-terminal session with the SCC GUI.

Data Stripper Controller Node Interface *(c:\Master\DSCInterface.exe)*

The Data Stripper Controller Node (DSC) Interface is a multi-threaded Windows NT application that coordinates communication between the Master and node controlling the TSI Data Stripper. This process is responsible for forwarding the schedule requirements to the DSC node PC; no ephemeris is required. Communication between the Master PC and DSC node is accomplished using one 2-way TCP/IP socket connection Successful socket communication requires the following Windows NT *hosts* and *services* configuration file changes:

(1) Socket communication service identification in *c:\winnt\system32\drivers\etc\services*.

dscnode_recv   3015/tcp   # Data Stripper (remote) Controller message receive service

(2) Socket communication IP address, hostname and aliases in *c:\winnt\system32\drivers\etc\hosts*.

150.xxx.xxx.xx wgsmaster1   remote_host   thisone
128.xxx.xxx.xx datastripper   thatone

The alias, *remote_host*, is required alias for the ATS Master PC process *11mInterface*.   The alias *thisone* Is also used to identify the ATS Master PC to *the DSCInterface* application. The hostname for the datastripper and an alias of *thatone* are also required for the *DSCInterface* TCP/IP socket communications to the DSC node PC.

Successful delivery of the schedule file depends on the existence of a WS_FTP32 profile identified by the operator in the *c:\Master\Station\StationAssets.dat* file using the *StationAssetsEditor* application.   See **Chapter 4 – Configuring the Master for a Station. Starting the Station Assets Editor**.

The socket is initialized in order to send messages from the Master to the SCC.   These messages include:

(1) A request for socket connection.
(2) A request for remote login using the DSC node name account *remoteuser*.

The above four messages are delivered when the *DSCInterface* process begins execution. The following three messages are delivered at support *Initialization* time (usually AOS – 10 minutes) and after preparation and FTP of the remote schedule (*c:\Master\Schedule\DscNodes.scd*) by the *MonitorAndControl* process:

(1) A request for *control upgrade* status in order to prepare for loading the schedule files.
(2) A request to *load* the remote schedule file.
(3) A request for *control downgrade* status in order to release remote control back to the DSC node.

The TCP/IP socket connection is also responsible for listening and receiving instrument status packets, as well as granting approval for *login* and *control upgrade* requests.

Successful startup of this process requires prior startup and execution of DSC node processes and is marked in the *MessageLog* window as:

(04/06/1999 (096) 14:39:20) The DSC Interface process has been started.
(04/06/1999 (096) 14:39:20) DSC Interface... Send socket initialized for DSC_Host at WGS .
(04/06/1999 (096) 14:39:28) DSC Interface... Listen socket initialized for DSC_Host at WGS .
(04/06/1999 (096) 14:39:33) DSC Interface... Sending login request to DSC Node XIFSOCK.
(04/06/1999 (096) 14:39:33) DSC Interface... Remote login to DSC system approved.

Post-Pass Summary Results Compiler (c:\Master*PassResultsCompiler.exe)*

The *PassResultsCompiler.exe* marks the final stage of the automation cycle. It gathers information and statistics from tracking station resources and compiles a report summarizing the performance of the tracking station in support of a pass. It is also responsible for some general system administrative activities – like removing 30-day-old Master logs, schedules, SCC passlogs, pass result files, equipment operation logs and any other file residing in *c:\Master\Schedule\Archive*.

PassResultsCompiler is started approximately 90 seconds after scheduled LOS in order to respect the various delays encountered in the *Stop* and *Takedown* phases of the automation cycle and allow adequate time for all log files to be retrieved via FTP.   The 90-second wait period is ignored and replaced by a 10-minute timeout for supports which request PTP or Data Stripper Controller (DSC) node automation.

ATS will idle for a maximum of 10 minutes or until the PTP and DSC log files are retrieved by the Master PC.

Successful startup of *PassResultsCompiler.exe* is displayed in the *MessageLog* window as:

(03/24/1999 (083) 13:42:49) Attempting to takedown QST_03_24_1999_13_38_19_00...
(03/24/1999 (083) 13:42:49) The PassResultsCompiler process has been signaled to start.
(03/24/1999 (083) 13:45:49) Scheduler... PassResultsCompiler started for operational profile
QST_03_24_1999_13_38_19_00 .
*(03/24/1999 (083) 13:45:51) Scheduler... ***** Post-Pass Summary Complete (1 events, 1 S-band*
*events and 0 Xband events) *****

The following files are typically retrieved and processed by *PassResultsCompiler*:
SCC Files:

/users/aaas/etc/log/reports/satID.orbit-number.day-of-year.hour.minute (passlog file)
/users/aaas/etc/tapeentry.log (tapelog file; for X-band recording supports only)
/users/aaas/etc/message.file

Node Files:	Master Archive Folder
c:\Node\OperationLogs\MetrumBVLDSunit#OperationalProfileName	c:\Master\MetrumBVLDSTapes
c:\Node\OperationLogs\Avtec1001unit#OperationalProfileName	c:\Master\PtpLogs
c:\Node\OperationLogs\WffTdfunit#OperationalProfileName	c:\Master\TdfLogs

Node files retrieved by the Master are copied into the *c:\temp* folder during the process. After processing, they are copied to an archive folder and saved for thirty days.   This process automatically files which exceed 30-days in age.

Four files are updated during this process:
(1) *c:\Master\11MeterTapes\tapelog.dat*; an ASCII text file containing a list of 10-day-old (or newer) X-band tape recording data found in the binary SCC *tapeentry.log* file. This file is over-written and re-created when compiling pass results for an X-band support.
(2) *c:\Master\11MeterTapes\UnshippedTapes.lst*; an ASCII text file containing a list of all X-band tape names found in the binary *tapeentry.log* file which match the expected WOTRS scheduled event characteristics. These tape names are displayed using the utility *c:\Master\ShippingReport.exe*. Information is appended to this text file when compiling pass results for an X-band support.
(3) *c:\Master\MetrumBVLDSTapes\tapes.lst*; an ASCII text file containing a list of all S-band MetrumBVLDS tape recording data found in operation logs. Information is appended to this text file when compiling pass results for an S-band recording.   This information may be required in a future release of ATS to support automated playback requirements.
(4) *c:\Master\MetrumBVLDSTapes\UnshippedTapes.lst*; an ASCII text file containing a list of all S-band MetrumBVLDS tape names found in operation logs which match the expected WOTRS scheduled event characteristics.

Two files are created during this process:
(1) The pass result summary text file is created at *c:\Master\Schedule\SIFx* (where x represents either 'W' for Wallops, 'A' for AGS, 'M' for McMurdo or 'S' for SGS.)   The pass result files are named according to the following convention:

PRFxyyyymmdd_OperationalProfileName;
    where x   = character 'W' for Wallops, 'A' for AGS, 'M' for McMurdo and 'S' for SGS
           yyyy   = 4-digit year
           mmdd   = 2-digit month and 2-digit day

    The format and contents of the pass results report are defined in

Appendix C: Pass Results Format**.**
These files are automatically deleted after thirty days. This file is sent via FTP to WOTIS using the WS_FTP32 profile contained in *c:\Master\Station\StationAssets.dat*.

(2) The Universal Tracking Data Format (UTDF) files for scheduled supports that require automation of the Tracking Data Formatter (TDF) device are created at *c:\Master\Utdf* and saved for thirty days. This file is sent via FTP to UTDF customers using the WS_FTP32 profile contained in the operational profile. These files follow two naming conventions:

      (a) Landsat7-specific files; *L7yyyydddTRKsid.V##*,

            where,   yyyy   = 4-digit year
                     ddd   = 3-digit day-of-year
                     sid   = 3-character siteID ('WPS' for Wallops, 'SGS', 'AGS' or 'MGS')
                     ##   = 0-based version number to accommodate multiple files on same day.

      (b) all other UTDF files, SSUTDF_SupportIDVehicleID_RcvPadID_yyyy_ddd_hhmm.trk

where, SS   = speed, or rate of data report; 'HS' for high-speed (10 points/second) or 'LS' for
                       low- speed (1 point/second);
     SupportID = 4-digit support identifier;
     VehicleID = 2-digit vehicle identifier;
     RcvPadID = 3-digit receive pad identifier;
     yyyy   = 4-digit year;
     ddd   = 3-digit day-of-year;
     hhmm   = 2-digit hour and 2-digit minute

SAFS Heartbeat (*SAFSHearbeat.exe*)

SafsHeartBeat.exe is a core process started automatically by *M&C*. It idles forever, only interrupted every two minutes to detect the local Standard Autonomous File Server (SAFS) system integrity via receipt of the TCP/IP socket message "sid BEAT", where (where sid represents either *WPS*, *AGS* , *MGS* or *SGS*). Startup of this process in displayed in the *MessageLog* window as:

(03/23/1999 (082) 23:32:45) The SAFS Interface has been started.

Any failure to detect this process is displayed in the *MessageLog* window as:

(03/23/1999 (082) 23:45:08) SAFS Heartbeat... Timeout on receive of SAFS heartbeat

The SAFS can be identified as a critical ATS resource for supports that require automation of the Programmable Telemetry Processor (PTP) using the *ProfileEditor* utility.

Utilities

Some of the utilities require an authorized user. An authorized user is a user that knows the Master subsystem password. Those processes that <u>do</u> require an authorized user are marked with [*]. The Master includes the following utilities:

- Station Assets Editor[*]
- Profile Editor
- Master Password Editor[*]
- Station Status Display
- Shipping Report Writer

23

- Manual Schedule Notification

A brief description of each of the pre-support utilities is provided here; specific instructions for using each of the utilities can be found in later chapters of this manual or by accessing the application's on-line help documentation.

Station Assets Editor *(c:\Master\StationAssetsEditor.exe)*

The Station Assets Editor is a utility with which the station specific characteristics of the tracking station can be described. It can be used to identify the automated antennas controlled by the tracking station, specify the destination of pass result and tape shipping files, register the Master workstations associated with the tracking station, and define communication protocols used by the tracking station. For more information about the Station Assets Editor, see **Chapter 4 -Configuring the Master for a Station**.

Profile Editor *(c:\Master\ProfileEditor.exe)*

The *ProfileEditor* is one of the most important tools provided by ATS. It allows operations personnel to identify ground station resources that will be cycled through automation to accommodate support requirements. This utility allows selection of resource requirements, SCC configuration ID and UTDF clients. It also allows the user to specify the support *Initialization*, *Setup* and *Start* times prior to the scheduled AOS. There are two types of profiled which can be edited by this utility:

(1) *Base* profiles are satellite-specific folders contained at *c:\Master\Profiles* which contain information regarding the types of resources and associated configuration files that support a scheduled request. These profiles are created by the operator from information contained in a project Internal Control Document (ICD). *Base* profiles follow no specific naming convention and are deleted only by manual interaction. Information in base profiles is matched to similar data in the scheduled request during schedule ingestion in order to create the operational profile. *Base* profiles are editable at anytime, but changes are only integrated for future, or upcoming, schedule deliveries.
(2) *Operational,* or *support,* profiles are created automatically by ATS from the best-fit base profile during schedule ingestion. *Operational* profiles guide a specific support request through the automation cycle. These profiles are typically named according to the scheduled WOTIS satellite identifier and AOS and created at *c:\Master\OpProfiles*.   They can be edited anytime and changes are integrated immediately into ATS provided the support has **not** started and passed through *Initialization*. Support profiles are automatically deleted three days after their support concludes. Receipt of a new schedule **will not** destroy and over-write *operational* profiles.

For more information about the Profile Editor, see **Chapter 5 : Profiles**.

Master Password Editor *(c:\Master\MasterPassword.exe)*

Access to some Master functions is limited. This is to prevent the inadvertent or mistaken execution of an option that may interfere with the Master's ability to conduct scheduled support activity. Therefore, the user may be prompted to enter a password before the Master will carry out a command. The password that is required is a Master subsystem specific password defined by an authorized user. (Note: this password is not related to the user password required for login). The Master Password Editor can be used by authorized users to change the Master subsystem password. Knowledge of the current password and confirmation of the new password is required.

Station Status Display *(c:\Master\StationStatusDisplay.exe)*

The *StationStatusDisplay* application is a utility that displays the TCP/IP socket status message contents transmitted by the broadcast thread started by *MonitorAndControl.*   This utility is available to any remote client interested in viewing station status available during a support. Requirements to execute this utility include:

(1) Windows NT 4.0 PC;
(2) Closed-network connection capability to the ground station Masters.
(3) The registration of the client PC into the Master using the *ProfileEditor* application.

Although *StationStatusDisplay* receives status from all ground station, only one is viewable at anytime.

Tape Shipping Report Writer *(c:\Master\ShippingReport.exe)*

The Shipping Report Writer utility is a dual-purpose application. It allows a user the capability to:
(1) **Prepare X-band tape shipping report files**. A tape shipping report is a text file created in the *c:\Master\Schedule\SIF** folder (where * represents 'A', 'M', 'S' or 'W' for AGS, MGS, SGS or the Wallops ground station, respectively). The format and contents of the shipping report are defined in **Appendix G: Tape Shipping Report Format.** Operator input is limited to the selection of the tapeID(s) for shipment and the 10-character box, or conveyance ID. The ground station name ('WPS', 'MGS', 'SGS' or 'AGS') and recorder type ('AA' for Ampex and 'SS' for Sony) are automatically inserted into the 15-character tapeID name. A list of selectable tapes is compiled during pass results compilation at support takedown. This list is archived and updated at *c:\Master\11MeterTapes\UnShippedTapes.lst*. The tape shipping report file is automatically transferred via FTP to the directory and machine identified in the *PassResultsDestination:* and *PRFRecipientFTPProfile:* fields in the editable station assets (*c:\Master\Station\StationAssets.dat*) file. These fields are typically defined for the Wallops Information System Data Base (WISDB) at Wallops. Tape shipping report files are created by the operator when tapes are boxed and ready for mail shipment to a customer. The WISDB is responsible for surveying old pass result files for evidence of reported tape recordings upon receipt of a tape shipping report file and sending a notification message to the tape customer.
(2) **Prepare duplicate X-band tape report files**. An operator-assisted tape duplication procedure is required by ground station personnel when real-time recordings are not possible. Automatic FTP of this file to the WISDB will force the data base to archive the duplicate tape name along with the primary one. Operator input is limited to the selection of the tapeID used for the duplication and a 10-character field for it's name. Both fields are entered into a duplicate information text file at *c:\Master\Schedule\SIF** . The file is transferred to the same destination at WISDB which receives shipping report files. See **Appendix K: Duplicate Tape Report Format** for file contents.

Manual Schedule Notification *(c:\Master\ManualNotification.exe)*
The Manual Schedule Notification utility provides a tool with which a user can issue a new schedule notification message to the Master. This notification signals the *Scheduler* process to re-ingest the WOTIS-delivered schedule file (*c:\Master\Schedule\wotrs.mas*), as well as any user-inserted, modified or deleted events referenced at *c:\Master\Schedule\UserModifications*. *ManualNotification* allows operations personnel to logoff the Master during extended idle times without worry that a WOTIS schedule will be missed. WOTIS schedule files will continue to be delivered via FTP to the Master without execution of ATS.
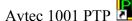
Device Interfaces

A device interface includes a GUI and device driver and command classes. Device interfaces are available for the following devices:

- Avtec 1001 Programmable Telemetry Processor
- Aydin 329A BPSK Demodulator
- Datatape 3700J Analog Tape Recorder
- Decom Systems 7715 Bit Synchronizer
- General Data Products GDP 225D Frame Synchronizer
- General Data Products GDP 233 PCM Simulator

- General Data Products GDP 783M BPSK Subcarrier/Modulator
- Hewlett Packard 3325B Synthesizer
- Hewlett Packard E1366/67A RF Matrix Switch
- Krohn-Hite 3905B Programmable Filter
- Metrum BVLDS Cassette Digital Tape Recorder
- Optrax SS100-B Digital Matrix Switch
- Optrax SS300-3S Analog Matrix Switch
- Matrix MSC10693 Analog Matrix Switch
- General Data Products GDP 911 Digital Matrix Switch
- Scientific Atlanta 11-meter Antenna
- WFF Telemetry Data Formatter (TDF)
- Apogee2208 Telemetry Data Formatter

A brief description of each of the device interfaces is provided here; specific instructions for using each of the interfaces and the various access modes can be found in later chapters of this manual or by accessing the application's on-line help documentation.

Avtec 1001 PTP

(c:\Master\ProgTMProcAvtec1001.exe, c:\Node\ProgTMProcAvtec1001.exe)

The Avtec Programmable Telemetry Processor (PTP) is a PC-based, multi-channel telemetry and command processing system. The PTP is used as a telemetry front-end system that performs data acquisition, real-time network transfer, and store and forward operations. Support for both time-division multiplexed (TDM) and CCSDS telemetry formats provides the flexibility to support multiple spacecraft with a single front-end system.

The PTP acts as a gateway that accepts multiple telemetry streams and outputs time-tagged frame or packet data over a network to workstations in a distributed satellite control and analysis system. The PTP also includes a command gateway that accepts input from the network and outputs serial commands to the up-link. Details on PTP use, automation requirements, and procedures are in **Appendix E: Programmable Telemetry Processor (PTP) Automation.**

Aydin 329A BPSK Demodulator

(c:\Master\DemodulatorAydin329A.exe, c:\Node\DemodulatorAydin329A.exe)

The Binary Phase Shift Key (BPSK) demodulator separates coherent or non-coherent PSK-NRZ, RZ or split phase modulated data from an FM multiplexed signal. It performs subcarrier detection and synchronous demodulation of an input PSK modulation channel contaminated by wide band noise and imbedded within a multiplex of other PSK and FM channels.

This device is used as part of the telemetry data path from the antenna to the data handling equipment. Output from this device is directed to a bit synchronizer. The Demodulator is connected to other devices in the telemetry signal data path by way of its signal input/output ports. The device's inputs/outputs are switched through an analog matrix switch. Moreover, the command-input port of this device is connected to its node computer through a Digiboard switchable serial port controller.

Decom Systems 7715 Bit Synchronizer

(c:\Master\BitSynchronizerDecom7715.exe, c:\Node\BitSynchronizerDecom7715.exe)

The bit synchronizer is a device used to synchronize and reconstructs a serial PCM bit stream which may be contaminated by noise, phase jitter, amplitude and baseline variations. The device provides as outputs reconstructed NRZ-L data, coherent clocks and various status signals.

General Data Products 225D Frame Synchronizer

(c:\Master\FrameSynchGDP225D.exe, c:\Node\FrameSynchGDP225D.exe)

The Model 225D is a PCM Frame Synchronizer unit that is used to acquire PCM Telemetry Data at a minor or major frame rate. It accepts serial data and clock then provides reconstructed data samples to a parallel output connector along with word number, frame number, major and minor frame status, and timing strobes.

General Data Products 233 PCM Simulator

(c:\Master\PCMSimulatorGDP233.exe, c:\Node\PCMSimulatorGDP233.exe)

The PCM simulator provides a serial PCM data stream at the same rate and code type with the same format characteristics - frame length, frame sync pattern, bits per word, words per frame, etc. - as a spacecraft data to be acquired by the ground station. This simulated spacecraft data stream is then used to verify the ground station operation before the spacecraft acquisition, i.e., pre-operation verification.

The critical tests to be made with the simulator are bit synchronizer and frame synchronizer acquisition of the simulated data stream. The PCM simulator output can be routed via the switch matrices directly to any of the sixteen bit synchronizer inputs for individual equipment tests or to test inject for modulation of a RF carrier to test the entire station acquisition including the antenna, receivers, bit synchronizers, frame synchronizers, WFEP, etc. Intermediate tests such as modulation of a simulated subcarrier with PCM data to test a subcarrier demodulator with a bit sync and frame sync can also be made.

General Data Products 783M BPSK Modulator

(c:\Master\ModulatorGDP783M.exe, c:\Node\ModulatorGDP783M.exe)

The General Data Products 783M BPSK modulator includes four BPSK modulators, a digital signal interface and a signal summer/level control module.

Each of the individual modulator sections contains a tunable subcarrier frequency generator, an internal pseudo-random-noise (PRN) generator, and a digital BPSK modulator. The GDP 783-M accepts a variety of input sources to feed its modulators. The 783-M accepts RS-422 level command data on four different lines, as well as an external analog signal. The internal PRN data can be selected in place of the external data inputs. The external inputs are first connected to the digital interface module.
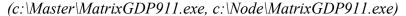
The four PSK modulator outputs, along with the two Digital Baseband signals, and the external analog input signal then can be connected to a summing section to create a composite output signal. Up to four of the seven signals may be selected for the summer.

Independent power level adjustment and on/off control is provided on the summer for each of the four signals that are inputs to the summer, as well as selectable attenuation of the composite output. The composite output signal will be in the form of a sine wave.

Four PSK modulator outputs are also provided as separate outputs on the rear panel separate from the feeds that may be selected for routing to the summer. Independent on/off selection and power level adjustment for these independent signals are provided on each modulator.

This device will be used as part of the command up-link path from the data handling equipment into the antenna. This device will be connected to other devices in the telemetry signal data path by way of its signal input/output ports. This device's input/output ports will be switched through an analog matrix switch. Moreover, the command-input port of this device will be connected to its node computer through a Digiboard switchable serial port controller.
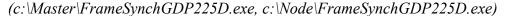
General Data Products 911 Digital Matrix Switch

(c:\Master\MatrixGDP911.exe, c:\Node\MatrixGDP911.exe)

The GDP-911 is a three-stage coaxial 64 x 64 matrix switch. The three stages are comprised of an input stage, a central stage, and an output stage. The switch's role in the automated tracking station is to route signals from inputs to outputs. Any input can be routed to one or more outputs. It can also be used to route test data streams. Like all other automated devices, a pre-existing configuration file can be loaded or created, and saved. The configuration file is loaded automatically by ATS at initialization time. This switch GUI – like the MSC10693 analog device - is composed of three views and/or child frames: (1) Control view is used to specify the input and output connections. The user is allowed to load a configuration file through the use of a Load New button. At this point the operator can identify either zero or one source for each of the sixty-four targets or verify the contents of the pre-existing file and modify. Multiple targets can share the same source. To set the connections, click on a source name adjacent to the desired target name and select the source name option or select "None" for no connection to the target from the drop down menu. When the operator executes a "Load New", chooses a file, and clicks open from the NT Explorer domodal box the configuration file will open. As the file opens and the data is applied to the control view window, it is concurrently being sent to the device and displays the status of the device in the status view window. "Apply Now" will apply the connections and disconnections (if applicable) and must be clicked after every "Load New". (2) Settings views is used to specify the source and target names. These names are saved in *c:\Master\Station\MatrixGDP911IO.dat* . The left column contains the current source names and the right column shows target names. To edit a name, select it from the appropriate list control, then click on it to make the change; this edit feature is similar to the re-name convention used by Windows.
Press "Enter" to complete the edit. Selection of "Apply New Names" will save the new name settings. Click "Reset Names" to restore the previously-saved names (new names will not be saved). (3) Status view is a "read-only" window which displays the current device connections.

Hewlett Packard 3325B Synthesizer

(c:\Master\SynthesizerHP3325B.exe, c:\Node\SynthesizerHP3325B.exe)

The Hewlett Packard (HP) 3325B Synthesizer/Function Generator produces sine wave, square wave, triangle waveforms and positive and negative ramp waveforms. It is used to set the reference frequency for the tape recorders. TTL will be supplied to the digital recorders and sine wave will be supplied to the analog recorders.

Hewlett Packard E1366/67A RF Matrix Switch

(c:\Master\MatrixHPE1366A.exe, c:\Node\MatrixHPE1366A.exe)

The Model E1366A 50 ohm RF Multiplexer cards are used to route low level as well as high level analog and RF signals that must operate without distortion or interference over a very wide frequency range. These switch banks provide the required isolation to the adjacent channels that is not available in a cross point matrix. Four cards are mounted in a E1301A mainframe for control and cables are connected to make the proper signal routing that often requires routing through multiple switch banks so that alternate signals can also be routed to the same locations.

For real-time support, the inputs selected by the switch banks are PTP commands received from NASCOM via IP transmission. These switch banks route the command signals to the PSK modulator input. The PSK modulator output is routed by these switch banks to the Up-link Control Unit that is connected to the exciter in the antenna pedestal. Switching allows selection of different PTP units, different PSK modulators and different Up-link Control Unit channels as needed.

For pre-operation verification of the ground station equipment the inputs selected by these switches may be the bit error rate test set, the PCM simulator, the Forward Error Correction Encoders in the bit synchronizers or any of the PTPs. These inputs may be routed to the PSK modulator input, directly to the high rate channel input of the Up-link Control Unit or to the test inject signal generator in the antenna pedestal. The test inject signal generator is also selectable as an up-link exciter. These switch banks also provide routing to the analog matrix to permit testing of individual downlink equipment.

Krohn-Hite 3905B Programmable Filter

(c:\Master\FilterKrohnhite3905B.exe, c:\Node\FilterKrohnhite3905B.exe)

The Krohn-Hite Model 3905 mainframe with Model 34A and Model 35/36 plug-in filters provide two tunable filters. The primary use of these tunable filters is to remove unwanted sidebands from the PCM signal before it modulates either the test inject signal generator or the up-link exciter. The input of the Model 34A filter is permanently cabled to the Up-link Control Unit output so that it can only be used for filtering up-link command signals. The output of the Model 34A is an input to the command and test switch matrix. Its output or the output of the Model 35/36 can be switched to the input of the Fiber Optic modem which provides the input to the test inject signal generator or the up-link exciter in the antenna pedestal. The input to the Model 35/36 filter is selectable via the command and test switch matrix to permit a wide range of test signal inputs from the PCM simulator, Bit Error Rate Test Set, PTP, etc.

Metrum BVLDS Cassette Digital Tape Recorder

(c:\Master\RecorderMetrumBVLDS.exe, c:\Node\RecorderMetrumBVLDS.exe)

The Metrum BVLDS is a digital magnetic-tape drive that is used to capture and store real-time scientific data. It can store up to 10.4 gigabytes of data (with error correction) on a 120-minute tape cassette. When storing data at maximum rate and operating in two-channel mode, a T-120 tape will store a minimum of 43 minutes of data. It uses rotary helical-scan technology and standard commercial tape cassettes, much like consumer VCRs. The BVLDS can be configured with one or two channels.

Units that are used in automated tracking stations are configured with a single channel. Single channel units have data provided to the recorder buffer(s) at a rate of between one kilobit and thirty-two megabits per second. The transfer of data from a buffer to tape for a single channel unit is between one megabyte and two megabytes per second.
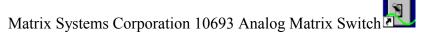
This device is attached to a Metrum specific serial controller, which is, in turn, attached to a digiboard general serial controller. Its data ports are connected to the matrix switch.

Optrax SS100-B Digital Matrix Switch

(c:\Master\MatrixOptraxSS100B.exe, c:\Node\MatrixOptraxSS100B.exe)

The Optrax SS100-B is a 16 X 16 digital matrix switch. All paths consist of two separate channels, clock and data, which are always handled in pairs by the switch controller. The switch's role in the automated tracking station will be to connect bit synchronous data and clock signals to NASCOM, recording devices, data processing, and test devices. Input from one channel can be routed to one or more output channels.

Matrix Systems Corporation 10693 Analog Matrix Switch

(c:\Master\MatrixMSC10693.exe, c:\Node\MatrixMSC10693.exe)

The MSC-10693 is a three-stage coaxial 32 x 32 matrix switch. The three stages are comprised of an input stage, a central stage, and an output stage. The switch's role in the automated tracking station is to route signals from inputs to outputs. Any input can be routed to one or more outputs. It can be used to route test data streams. Like all other automated devices, a configuration file can be created and saved. The configuration file is loaded automatically by ATS at initialization time. This switch GUI – like the GDP-911 digital device - is composed of three views and/or child frames: (1) Control view is used to specify the input and output connections. The user is allowed to identify either zero or one source for each of the thirty-two targets or verify the contents of the pre-existing file and modify. Multiple targets can share the same source. To set the connections, click on a source name adjacent to the desired target name and select the source name option or select "None" for no connection to the target from the drop down menu. When the operator executes a "Load New", chooses a file, and clicks open from the NT Explorer domodal box the configuration file will open. As the file opens and the data is applied to the control view window, it is concurrently being sent to the device and displays the status of the device in the status view window. "Apply Now" will apply the connections and disconnections (if applicable) and must be clicked after every "Load New". (2) Settings views is used to specify the source and target names. These names are saved in *c:\Master\Station\MatrixMSC10693IO.dat* . The left column contains the current source names and the right column shows target names. To edit a name, select it from the appropriate list control, then click on it to make the change; this edit feature is similar to the re-name convention used by Windows.
Press "Enter" to complete the edit. Selection of "Apply New Names" will save the new name settings. Click "Reset Names" to restore the previously-saved names (new names will not be saved). (3) Status view is a "read-only" window which displays the current device connections.

WFF Telemetry Data Formatter

(c:\Master\WFFTDF.exe, c:\Node\WFFTDF.exe)

The telemetry data formatter (TDF) is responsible for processing antenna angle inputs, as well as Doppler ranging, and timing information, in order to produce positional data products for use by other range of worldwide tracking assets. Time series data are output in the Universal Tracking Data Format (UTDF) via an RS-422 port and the Minimum Delay Data Format (MDDF) over an RS-232 line.

Chapter 3 A Guide to the Graphical User Interfaces (GUI)

Many of the software applications developed for the automated tracking station include GUIs. A GUI includes a window, or collection of windows, displayed on the computer screen that provides a user with a mechanism for interacting with the software to issue commands, select options, and enter input. Additionally, the software is utilized by operations and accessed by various modes (Chapter 7), and uses GUIs to present status data to the user. One of the goals in developing the user interface portions of the automation software was to maintain a consistent look and feel across applications. In general, the user interface applications have been developed with an attempt to maximize their usefulness to the user while minimizing the interaction required by the user. Every attempt has been made to make the user interface applications straightforward, intuitive, and friendly. Thus, while the user interface software has been developed by a variety of software developers with input from a diversity of potential end-users for a wide range of devices, there are some general guidelines that apply to all applications.

Menus

A menu bar is a horizontal list of items that represent an application's commands. The menu bar typically contains one or more menu items a user can select to display pull-down menus and is positioned at the top of the window, under the title bar (see **Figure 3-1**[1]). The menu bar may vary depending on the mode of the application.

Applications that manage multiple windows may create window specific menus for each child window.

Moving and Resizing

Windows can be repositioned by positioning the cursor over the top border, holding down the left mouse button, and dragging the window to the desired position. Most windows can be resized by positioning the cursor over the edge of the window, clicking and holding the left mouse button, and stretching the border to the desired size. Most windows also include a combination of window system buttons in the upper right hand corner of the window.

Minimize Button

The Minimize button shrinks the window to its minimum size.

Maximize Button

The Maximize button expands the window to its maximum size. A maximized window fills its container window or the desktop window.

Restore Button

The Restore button restores a maximized window to its previous size.

Close Button

In general, the Close button closes the window. In cases where the window being closed is the main application window, it also terminates the application. However, in some applications where it would be inappropriate to close the window containing the close button, the window will be minimized instead.

Toolbars

Some user interface applications provide a toolbar as a convenient access to commands. A toolbar is a row of buttons that execute application specific commands. The toolbar options may vary depending on the mode of the application. The toolbar is usually positioned directly beneath the menu bar (see **Figure 3-1**).

---

[1] The window in Figure 3-1 is from the interface to the Metrum BVLDS cassette digital tape recorder.

Status Bar

A status bar is a row of text panes that display status indicators. Most user interface applications provide a status bar along the bottom border of the application window (see **Figure 3-1**). In the lower left corner, the status of the application is reported. In the right corner, the status of the caps lock, number lock, and scroll lock keyboard options are indicated.

Figure 3-1 A Typical Interface Window

Title Bar

The title bar displays the device name and for some of the devices it also displays the working directory mode.

Dialog Boxes and Child Windows

A dialog box is a window that is typically displayed in response to an option exercised by the user or a situation encountered by the software. A child window is a window that is included in and contained in an application's main window area. These windows are used as both a means of presenting information to the user and as a method of soliciting input from the user. In the ATS software the program accesses the child frames that were created by the multiple document interface (MDI) wizard and specifically called upon by the GUI to select configuration files. The windows associated with the GUI are the Control View, Status View, and the Settings View.

Each of these types of windows may contain a variety of *controls*. A control is a graphical tool used by the user to issue commands or specify inputs to an application. A description of the typical controls employed by user interface applications is provided here; specific instructions for using the controls as they are implemented by each of the interfaces can be found in later chapters of this manual or by accessing the application's on-line help documentation.

Button

A button control is used to execute commands. They are often located near the bottom of dialog boxes and windows and used to apply parameters entered in other fields and/or close the window. Buttons may be enabled or disabled depending on the current mode of operation of the application. Buttons may be labeled with text or graphics, or they may be shaped to reflect their purpose. **Figure 3-2**[2] depicts a typical window utilizing a variety of controls; it includes a Load New and an Apply Now button[3].

---

[2] This window is from the Krohn-Hite 3905B programmable filter device interface application.
[3] These buttons are disabled in this figure.

Figure 3-2 Some Typical Input Controls
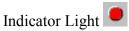
Check Box

A check box is used to choose from a set of related but independent options. A checked check box contains an **x**. Repeatedly checking a check box toggles it from state to state. A check box has two states: checked and unchecked. The GPIB Service Request ON control in **Figure 3-2** is a check box[4].

Combo Box/Drop-list

A combo box or drop-list presents the user a list of selectable items when the user clicks the arrow next to the control. The user can not edit the list of selectable items. In **Figure 3-2**, Pre-Filter Gain is specified using a drop-list.

Edit Box

An edit box allows the user to enter and edit text. These controls can be used for character or numeric fields. The window depicted in **Figure 3-2** includes an edit box for specifying Cut-Off Frequency.

Indicator Light

An indicator light is a graphical control used to reflect the status of a monitored parameter. The light usually has two states: OK (green) or error (red). This type of control accepts no interaction with the user. **Figure 3-3**[5] uses indicator lights to show the status of Signal Level.

---

[4] The check box is not checked in this figure.
[5] This window is from the interface to the Decom Systems 7715 bit synchronizer.

Figure 3-3 Some Typical Status Controls

List Box

A list box is a control that presents the user with a list of items or information. The effect of selecting items in a list box depends on the application.

Radio Buttons

A radio button is a round button operated by the user to choose from a set of related but mutually exclusive options. The check mark for a radio button is a black dot. A radio button can assume two states: checked (dot) or unchecked (no dot). In **Figure 3-2** the user specifies Channel by selecting the appropriate radio button.

Scroll Bar

A scroll bar is a sliding control that a user can manipulate to shift the contents of a window's client area either horizontally or vertically. It is usually positioned along the left and/or bottom window border when the window's client area is greater than the size of the displayed window. Scroll bars are visible along both the left and bottom border of **Figure 3-3**.

Spin/Up-down Arrows

A spin or up-down control includes a pair of arrows that the user can click to adjust a value displayed in an adjacent edit box. Clicking the up arrow increases the value, clicking the down arrow decreases the value. When the value is expected to contain a decimal value, the user has the option of changing a selected digit of the value. If a particular digit is selected, the increment and decrement associated with the up and down arrows will correspond to that digit. If no digit is selected, a default increment and decrement is used. A spin control is shown in **Figure 3-2** next to the Cut-Off Frequency field.

Static or Read-Only Text

Labels or status data are displayed as static or read-only text. These fields are used to convey information to the user and accept no input from the user. In **Figure 3-3** a read-only field is used to report the Bit Rate.

Switch

A switch control is a button designed to look like a light switch. A switch has two positions, up and down, that are toggled when the switch control is pressed. It is used to turn an option on and off.

Pendulum

Many status windows include a pendulum that "swings" to indicate that the status displayed in the window is being updated. In other words, the pendulum serves as a heartbeat monitor for the window. In **Figure 3-3** a pendulum is located in the center of the dialog.

Error-Reporting

The Master subsystem processes employ two different methods of error reporting. The Core Processes primarily report error information to the user via the Monitor and Control process's Message Log window. Errors encountered in the Utilities and Device Interfaces are reported to the user via a Message Box pop-up window. In each case, the description of the error indicates the name of the application, the nature of the error, and other information that may be useful in resolving the problem.

On-line Help

All Master user interface applications are equipped with some level of on-line Help. In general, a menu option on an application's main window menu bar can be exercised to open a separate, independent on-line help window.

Chapter 4 Configuring the Master for a Station

One of the objectives in developing the automated tracking station software is to minimize any station specific aspects of the software in order to maximize the re-usability of the software at various tracking stations. Thus, whenever possible, station specific characteristics are loaded, at run time, from a user managed data file, *c:\Master\Station\StationAssets.dat*. A tool for making the manipulation of the file containing station specific characteristics convenient and straightforward was developed as part of the Master subsystem. This tool, the *StationAssetsEditor*, can be used to configure the Master PC for a specific station.

Starting the Station Assets Editor

To Start the *Station Assets Editor*, double-click on the application-shortcut icon found in the desktop folder labeled *The Master*.

To start the *Station Assets Editor* from the *Monitor and Control* process main menu, choose the Update Stations Assets option from the Tools pull-down menu.

Describing the Automated Tracking Station

Information describing the automated tracking station is entered in the Automated Tracking Station window. Specifically, this window is used to define station specific characteristics as they relate to communication between the Master and the scheduling office. See **Figure 4-1**. The characteristics of the tracking station that must be defined in this window are:

❑ **Station ID**
This is the identifier used by the scheduling office to discriminate between tracking stations. It is generally a three-character acronym that is related to the full name of the station. Use *WPS* for Wallops, *AGS* for Poker Flat and *SGS* for Svalbard.

❑ **WOTIS Pass Results FTP Profile Name**
This is the name of the File Transfer Protocol (FTP) profile residing on the Master PC which defines the FTP communication parameters from the Master to the pass results recipient. This FTP profile name is identified in the windows *WS_FTP32* utility.

❑ **WOTIS Destination Directory**
This is the directory which pass result summary files will be delivered to from the Master PC using the FTP profile name described above.

❑ **Pass Result Summary Prefix**
This is the prefix with which pass results files will be named before they are delivered from the Master to WOTIS. The prefix choices include *PRFW*, *PRFA* and *PRFS* for Wallops, Poker and Svalbard ground stations, respectively. These prefixes must have similarly-named folders resident beneath *c:\Master\Schedule*.

❑ **Shipping Report Prefix**
This is the prefix which tape shipping information report files will be named before they are delivered from the Master to WOTIS. The prefix choices include *SIFW*, SI*FA* and SI*FS* for Wallops, Poker and Svalbard ground stations, respectively.   These prefixes must have similarly-named folders resident beneath *c:\Master\Schedule.*

Figure 4-1 The Automated Tracking Station Window

Describing the Master Workstations

Information describing the master workstations is entered in the Master Workstation window. Specifically, this window is used to define station specific characteristics as they relate to the hardware and network configuration of the Master workstations. See **Figure 4.2**. The characteristics of the tracking station that must be defined in this window are:

❑ **Number of Master Workstations**
   This is the number of Master, regardless of their proximity to the tracking station.

❑ **Workstation Name**
   This is the workstation name or CPU name, for each of the Masters.

❑ **IP Address**
   This is the IP address for each of the Masters.

❑ **Number of Monitors**
   This is the number of monitors attached to each of the Masters.

Figure 4-2 The Master Workstations Window

Describing the Automated Antennas

Information describing the associated with the tracking stations is entered in the Automated Antennas window. Specifically, this window is used to define station specific characteristics as they relate to communication between the Master and the automated antennas. See **Figure 4.3**. The characteristics of the tracking station that must be defined in this window are:

❑ **Antenna Name**
A simple antenna name; for example, *SA11Meter* or *SCC11*. The name must contain less than ten characters. It is loaded into ATS, but not used in automation.

❑ **WOTIS-recognized Antenna Identifier**
An integer used by WOTIS to identify an antenna resource. Typically 10 and 11 for the Scientific-Atlanta 10 and 11 meter systems, respectively. See **Appendix B: Master – WOTIS Communication**.

❑ **Antenna Resource Type**
An ATS recognized antenna resource name. The name corresponds to an enumerated list maintained in ATS source code.

❑ **Number of Monitored Instruments**
The number of antenna instruments which will be individually monitored for high-level status. The *MonitorAndControl* process reserves memory for a maximum of twenty instruments.

❑ **Antenna System FTP Profile**
This is the name of the File Transfer Protocol (FTP) profile residing on the Master PC which defines the FTP communication parameters from the Master to the antenna control system. This FTP profile name is identified in the windows *WS_FTP32* utility.

❑ **Antenna System Schedule Destination**
The full directory name on the antenna system that operational schedule files will be delivered. This field is also used to identify the location ephemeris files will be delivered on ViaSAT, Inc. antenna systems. Typically, ATS delivers schedules and ephemris to the directory named /usr/people/jeffd/etc/remote on the station control computer (SCC).

❑ **Antenna System Configuration Directory**
The full directory name on the antenna system which contains a list of configuration names. The ATS *ProfileEditor* application expects this list in order to provide an operator the capability to identify configurations when creating profiles. The ATS *PassResultsCompiler* retrieves the system message log file during Post-Pass Summary compilation.

❑ **Antenna PassLog Source Directory**
The full directory name on the antenna system which contains the passlog files. A passlog file is created during by the antenna system during a support. The ATS *PassResultsCompiler* retrieves the passlog file at post-pass.

❑ **Antenna Tape Log Source**
The full directory name on the antenna system which contains the binary tapelog file. This file contains records of the last 1000 X-band tape recordings. The ATS *PassResultsCompiler* retrieves the tapeentry.log file at post-pass.

Figure 4-3 The Automated Antennas Window

Describing the Automated, Non-Antenna Resources

Information describing the automated, non-antenna resources associated with the tracking stations is entered in the Automated, Non-Antenna Resources window. Specifically, this window is used to specify the number of available units for each automated, non-antenna resource type. See **Figure 4-4**.

Figure 4-4 The Automated, Non-Antenna Resources Window

Chapter 5 : Profiles

Profiles define the configuration for and behavior during a scheduled support. They can be classified as either base or support (operational) profiles. Base profiles are listed on the Master at *c:\Master\Profiles* . Operational profiles are saved at *c:\Master\OpProfiles* . Base profiles contain equipment configurations for any satellite-specific support. For instance, there may be a QuikSCAT-1 and QuikSCAT-2 base profile to define two different configurations for a QuikSCAT support. Operational, or support, profiles are created automatically on the Master from the base profile when a new schedule is delivered from the WOTIS scheduling office. Support profiles are created by matching the satellite ID, TR code and antenna designator from the base profile with similar contents delivered in the WOTIS schedule. The operational profile is named according to the satellite ID and scheduled AOS and is loaded by the Master at a specified time prior to AOS. For instance, a support profile may be named *QST_10_06_98_11_59_59_11* to indicate a QuikSCAT 11meter scheduled support on Oct. 6, 1998 at 11:59:59 GMT.

Support profiles can be created and modified by a user in anticipation of support activity. Moreover, the user can modify a support profile for a scheduled up until that time at which the Master begins setup procedures for the support. Base profiles also can be edited at operator convenience, but will not create or change operational profiles until the next WOTIS schedule delivery.

Starting the Profile Editor

The *ProfileEditor* application can be started four ways. It can be started directly from the workstation desktop, two ways from *the MonitorAndControl* process main menu, and from the *MonitorAndControl* process Operational Schedule window. The method of starting the application depends on the type of profile to be edited. The first two methods start the Profile Editor in a mode that permits manipulation of non-operational profiles only. In order to start the Profile Editor such that operational profiles can be manipulated, it must be started from the Monitor and Control process Operational Schedule window.

To start the Profile Editor from the workstation desktop, double-click on the application-shortcut icon found in the desktop folder labeled The Master. To start the Profile Editor from the Monitor and Control process main menu, choose the Edit option from the Profile pull-down menu.
To start the Profile Editor from the Monitor and Control process Operational Schedule window, choose the Support Profile option from the Edit pull-down menu. **(Note: Information obtained in this section will need to be revisited and corrected to include updated processes used by ATS)**

Defining the Support

Information describing the support and the spacecraft is entered in the Support Definition window. See **Figure 5.1**. The characteristics of the support that must be defined in this window are:

❑ Satellite ID
This is the identifier used by the scheduling office to discriminate between satellites. It is generally a three-character acronym that is related to the full name of the spacecraft or mission.

❑ Support ID Code
This is a code used by tracking station operators to identify the support.

❑ TR Code
This is a code used by tracking station operators to identify the support configuration.

❑ Spacecraft Frequency – *Not used at this time*
   ❑ Uplink – *Not used at this time*
   This is the uplink frequency in kilohertz of the spacecraft.
   ❑ Downlink – *Not used at this time*
   This is the downlink frequency in kilohertz of the spacecraft.

❑ Operation
This is the type of operation to be performed by the tracking station.

❑ Ranging
This is a check box that, if checked, indicates that ranging should be performed in conjunction with the support.

❑ Type – *Not used at this time*
This collection of check boxes is used to specify the type of support that is required.

❑ Data line – *Not used at this time*
This drop list is used to select the data line to be used for the support.

❑ Data Format – *Not used at this time*
This collection of check boxes is used to select the data rate and format to be used for the support.

❑ RF Link – *Not used at this time*
This collection of radio buttons is used to select the RF link to be used for the support.

❑ Description
This is a text field used to describe the profile and provide information that may help the user in distinguishing the defining characteristics of the profile.

Figure 5-1 The Support Definition Window

Defining the Antenna Requirements

The antenna requirements, including configuration, setup, start, and stop times, are specified using the Antenna Requirements window. See **Figure 5-2**. The antenna configuration is selected from the Antenna Configuration window. . See **Figure 5-3**.

Figure 5-2 The Antenna Requirements Window

Figure 5-3 The Antenna Configuration Window

Defining the Resource Requirements

The resource requirements, including the type of resources, number of units, requested units, expected connections, resource setup, start, and stop times, and bit synchronizer dip switch settings are specified using the Resource Requirements window. See **Figure 5-4**.

- ❑ **Resources**
  Information describing the resource requirements for automated, non-antenna resources can, and should be, specified for all resources required for an anticipated support.
  - ❑ **Type**
    The information specified in the fields in this group pertains to, and only to, the resource type selected from this list. The user must go through the process of selecting type and specifying unit and time information for that type for all resource types needed for an anticipated support.
  - ❑ **Available Units**
    This is the number of units of the selected type that are available for automated supports. The information is presented for the user's information only; no modification of this information by the user is allowed.
  - ❑ **Required Units**
    This field is used to specify the number of units required for a support.
  - ❑ **Configure all units the same way**
    This check box is used to specify whether all units will share the same configuration or if each unit will have a unique configuration.
  - ❑ **Requested Units (1 – 10)**
    These fields are used to specify the units required for a support. The number of specific units identified must correspond to the number of required units. If all units are to be configured the same way, a configuration must be identified for the first selected unit only. Otherwise, for each specified unit, a configuration should be selected. Furthermore, if the selected unit is connected to one of the automated switches, the required target connections for that unit should be identified.
    - ❑ **Edit a configuration**

This button creates the device interface[6] corresponding to the selected type that can be used to specify detailed configuration data. . In this Remote Edit mode, you do not have access to Load New or Apply Now to manipulate the configuration file.

There are four ways to access the profile editor:

First, from the desktop. The operator can access the profile editor from the Node Style GUI Shortcut folder or from the Windows NT explorer. Double click on the icon. Select File->open to select a configuration file. The control view window appears with the chosen configuration file contents. The operator can make a change and save the file with the new changes.

Second, To edit a file from the Monitor and Control diagram, the operator selects Profile from the drop down menu and selects Edit . An open box will appear where the operator will need to select a configuration file and click okay. Upon the opening of the Profile Editor you will receive six windows: Support Definition, Antenna Requirements, Resource Requirements, UTDF Report, Station Status Broadcast Clients, Test Procedures. In the Resource Requirements Window the operator will select the type of device, select the number of units required, and the number of units/devices requested. And click Do It. The control view window appears with the chosen configuration file contents. The operator can make a change and save the file with the new changes.

Third & Fourth, To edit a file from the Monitor and Control diagram, the operator selects Profile from the drop down menu and selects Edit. An open box will appear where the operator will need to select a configuration file and click okay. Upon the opening of the Profile Editor you will receive six windows: Support Definition, Antenna Requirements, Resource Requirements, UTDF Report, Station Status Broadcast Clients, Test Procedures. In the Resource Requirements Window the operator will select the type of device, select the number of required units, the number of units/devices requested. The open box will appear for the operator to select a configuration file to Import from another profile or load a test configuration file, select the profile to be imported, and select Do It. This will invoke the initialization of the GUI with the opening of the control view window and its file contents loaded. The operator can make the selective changes and click save from the toolbar or from the drop down menu. The new file contents will be saved to the initial configuration file selected upon the operator invoking the profile editor the first time.

- ❑ **Setup (pre-pass seconds)**
  Not used. The devices are setup during the automation cycle immediately upon verification that initialization has succeeded.
- ❑ **Start (pre-pass seconds)**
  This is the number of seconds before scheduled AOS to be allowed for the selected type to start. Units of the selected type will be directed to start at a time calculated by subtracting the number specified in this field from the support start time. It is recommended that the start time be adequate for the resource to receive the command and respond to the Master. In operational testing, the Metrum BVLDS and PTP resources were the most sensitive resources to a proper start time, requiring 30 to 90 seconds.

---

[6] For more information about the device interfaces, see **Chapter 7 Using the Device Interfaces.**

❑ **Stop (post-pass seconds)**
This is the number of secondsafter scheduled LOS to be allowed for the selected type to stop. Units of the selected type will be directed to stop at a time calculated by adding the number specified in this field to the support stop time. In operational testing, no resources needed an adjusted stop time.

❑ **Bit Synchronizer Forward Error Correcting (FEC**)[7]
  ❑ Ignore dip switch settings. This check box is used to indicate whether or not the bit synchronizer dip switch settings should be considered in assigning a bit synchronizer to a support.
  ❑ Polarization - Inverted or Normal (SW1-7). This set of radio buttons indicates the correct bit synchronizer polarization dip switch setting for the spacecraft.
  ❑ Direction - Forward or Reverse (SW1-8). This set of radio buttons indicates the correct bit synchronizer direction dip switch setting for the spacecraft.

Figure 5-4 The Resource Requirements Window

Defining UTDF Recipients

The UTDF Recipients are the names of the FTP profiles residing on the Master workstation that defines the FTP communication parameters from the Master to the Universal Tracking Data Format (UTDF) recipient. The list of FTP profiles reflects those profiles defined for the local Master. See **Figure 5-5**.

Figure 5-5 The UTDF Report Window

---

[7] The FEC settings are made on dip switch SW1-7 and SW1-8 on the Bit Synchronizer Model 7715 CPU board. FEC is not used for ground station support on LandSat7 or EOS AM-1 so these switch settings will have no effect on scheduled support at SGS. FEC is used on nearly all Goddard satellites for the convention of G1G2 = Forward and G2 Inverted (SW1-7 = Off and SW1-8 = Off). The only exception supported by Wallops is the JPL satellite GEOTAIL which requires SW1-7 = Off and SW1-8 = On. The top cover must be removed to remove the CPU board to set the dip switches.

Chapter 6 Monitoring and Controlling

The *MonitorAndControl* (*M&C*) process is the primary interface between the user, the Master, Remote Nodes, and integrated independent systems. In other words, this process coordinates the monitoring and controlling of tracking station assets. It is one of the Master core processes and is essential to the successful automation of the tracking station.

As the primary interface between the user and the Master, the *MonitorAndControl* process behaves as a GUI. A wide range of options is available to the user with respect to the types and sizes of windows that can be displayed. As a rule, the *Tracking Station Resources Window*, the *Operational Schedule Window*, and the *Message Log Window* are always displayed. The *High Level Status Windows* are optional and can be activated, sized, and positioned according to the user's preferences.

A NASA ground station equipped with the Automated Tracking Station (ATS) software is typically setup with one Master PC and one or more remote node PCs. ATS can, however, be executed on one PC. The McMurdo ATS, for instance, is executed on one PC in order to satisfy space limitations.

What is the *Scheduling Master*?

In order for ATS to initialize node-controlled equipment that requires profile-requested antenna support, one Master PC must be dedicated as the *Scheduling Master*.   The *Scheduling Master* must have the following ATS core processes started in a successful state:
  (1) MonitorAndControl.exe
  (2) Scheduler.exe
  (3) Grm.exe
  (4) WotisInterface.exe
  (5) SAFSHeartbeat.exe
  (6) 11mInterface.exe

The operator is responsible for manually starting the *MonitorAndControl* application. This process is started automatically when user *Operator* logs in to the Master PC. It can also be started from the *Developer* login account by selecting the *MonitorAndControl* shortcut located on the user desktop. *MonitorAndControl* will attempt to automatically start the *Scheduler*, *Grm*, *WotisInterface* and *SAFSHeartbeat* applications. The *11mInterface* process is started upon user selection of *Yes* to the menu question "*Make This the Scheduling Master* ?" at *MonitorAndControl* startup. The *11mInterface* application can also be started from *the System->Start 11mInterface* menu after *M&C* is started. The user can also select "*Make This the Scheduling Master*" from the system menu. The ATS *Message Log* window contains errors associated with any core process The *SchedulingMaster* state is displayed on the block diagram as a light blue block surrounding the Master box.

ATS can also be placed in a *SchedulingMaster* state for testing or non-antenna support. This state can be reached if the Station Control Computer is down or not required for execution of the ATS *11mInterface* application. This state is useful when ATS profiles require a zero receive/transmit antenna for supports.

What is the Reporting Master?

One Master is responsible for reporting equipment status to remote clients using the *Station Status Monitor* thread started in *M&C* (see **Appendix F: Station Status Broadcasts**). These clients are identified in the base and support profiles. A keyword of "None" indicates no remote client identification. A maximum of three clients can be identified for any support. The ReportingMaster state is activated when support initialization is reached. It cannot be started manually.

Starting the Monitor and Control Process

The *MonitorAndControl* application is started automatically when an operator logs on to a Master workstation. However, in the event that the Monitor and Control process needs to be started manually, an application-shortcut icon is included in the desktop folder labeled The Master. To start the Monitor and Control process double-click the icon.

When the Monitor and Control process is started, it performs numerous steps in initializing the subsystem. It loads station specific data, gathers information from other Masters, initializes status data, establishes communication with the Remote Nodes, etc. This initialization process should take approximately one minute.

The Tracking Station Resources Window (TSRW)

The Monitor and Control process always displays the Tracking Station Resources. It presents a very high level status of the automated tracking station. The tracking station assets are presented as a block diagram of buttons graphically connected as the actual assets may be physically connected in the station. Each button represents a collection of a specific make and model of equipment or a Master workstation. The buttons indicate, by their border, the status of the units they represent.

The Master button borders may be green, red, light blue, or purple. A green border indicates that the Master workstation is reporting a heartbeat. This means that the Master has all basic core processes running; see **Figure 6-1 MASTER 3**. A red border indicates that the Master is not reporting a heartbeat. The Scheduling Master is indicated by a light blue border; see **Figure 6-1 MASTER 1**. The Reporting Master is indicated by a blue border; see **Figure 6-1 MASTER 2**. A purple border (not shown) indicates a Master is both the Scheduling Master and the Reporting Master.

For equipment buttons a green border indicates that all units of the equipment type represented by that button are reporting a healthy status. Conversely, if a button has a red border, at least one unit is reporting an error. For example, the button labeled BIT SYNC in **Figure 6-1** indicates that one of the bit synchronizer units in the tracking station is reporting a problem while all other buttons reflect a healthy status for all other equipment types.

Figure 6-1 The Tracking Station Resources Window

Pressing a button in the Tracking Station Resources Window activates a pop-up window containing more detailed status information for the associated equipment type. These *High Level Status Windows* list the high level status for each unit of an equipment type. The contents, format, and update rate for these windows are dependent on the equipment type.

The Operational Schedule Window

The Monitor and Control process GUI includes a window displaying the current operational schedule. The Operational Schedule window lists, in chronological order, all events scheduled by WOTIS as well as any user-defined events. For each event, a summary of the event characteristics is listed. See **Figure 6-2**. A complete list of an event's characteristics can be obtained by either double clicking on the event or by highlighting the event and choosing Event Details from the Operational Schedule Window's View pulldown menu. See **Figure 6-3**.

Figure 6-2 The Operational Schedule Window

Figure 6-3 An Event Details Dialog Window

Receiving a New Schedule from WOTIS

When a new schedule is delivered from WOTIS, the Operational Schedule window is automatically updated. No acceptance of the new schedule is required on the part of the user.

Editing an Operational Profile

With the Operational Schedule window Edit pulldown menu, the user has several options with which the operational schedule can be modified. However, the Master does not perform any conflict resolution; changes to the schedule made by the user may interfere with the accomplishment of the operational schedule delivered by WOTIS.

Adding new events to the Schedule

Using the Edit pulldown menu Insert Event option, the user can add events to the operational schedule. All pertinent characteristics of the event should be identified in the New Event dialog window, see **Figure 6-4**. In particular, a satellite ID, operation, acquisition of satellite (AOS), loss of satellite (LOS), start, and stop times are essential and must be included for all inserted events.

Figure 6-4 A New Event Dialog Window

Modifying scheduled events

The start and/or stop times for a scheduled event can be modified by pressing the Start button or the Stop button on the Event Details dialog window and entering new time values. The Start time must be no earlier than the AOS time; the Stop time must not be later than the LOS time. See **Figure 6-3**. The Event Details dialog window can be activated by one of the following methods:
- Double click on the event to be modified.
- Choose Event Details from the Operational Schedule Window's View pulldown menu.
- Choose Modify Event Start/Stop Times from the Edit pulldown menu.

Deleting events from the Schedule

Events can be deleted from the Operational Schedule. When an event is deleted from the schedule, the automated tracking station performs no functions related to the event. No report is made to WOTIS regarding the deleted event. The user can delete an event by highlighting the event to be deleted and choosing Delete Event from the Edit pulldown menu.

Viewing the Non-Automated Schedule

The schedule delivered to the Master from WOTIS is the full operational schedule for the tracking station and may include scheduled events for non-automated antennas. However, the Operational Schedule Window only displays those events scheduled for an antenna included in the automation effort. The user can view the *non-automated schedule* by choosing the option Open Non-Automated Event Schedule from the Operational Schedule Window's File pulldown menu.

Viewing the List of "Things to Do"

When the Scheduler receives an updated schedule, it generates a list of things that must be accomplished in order for the Master subsystem to accomplish the operational schedule. This list, containing a time tagged list of steps the Master will take, is printed to a file. The file can be viewed by choosing the option Things To Do from the Operational Schedule Window's File pulldown menu

The Message Log Window

The Message Log window displays error messages, status messages, and other useful information that is considered useful in monitoring the performance of the Master subsystem. It is a continuously scrolling window that is routinely updated whenever the Master is executing. See **Figure 6-5**.

Figure 6-5 The Message Log Window

Examining Log Files

Every message written to the Message Log window is also written to a *log* file. A log file is created by the Master each time the Monitor and Control process is started and saved each time the process is closed. In addition, as a log file reaches a size of 500 kilobytes, it is saved and a new one started. The log files are named according to their last save date and are located in the subdirectory **Log**. The Message Log window File pulldown menu Open Log option presents the user with a list of existing log files and starts Windows NT Notepad[8].

Logging User Messages

Since observations and comments made by a user may be useful additions to a message log, a utility with which a user can add messages to the message log is available. As with software generated messages, messages added by a user are time tagged and written to both the log window and the log file. To enter a message into the log, choose Insert Note from the Edit pulldown menu. This activates a very simple dialog in which the user can enter the text of the message to be logged.

Filtering Messages

Many of the messages logged by the software are generated as a reference for programmers who may need to perform troubleshooting. A user may wish to limit the display of messages to only those messages reporting errors and/or warnings that may indicate a potential failure of the Master to perform properly during scheduled support activity. This can be done by choosing Errors and Warnings Only from the View pulldown menu. This option operates as a toggle; the state of the filter is indicated by a check mark ✔ adjacent to the menu option text. When the filter is on, the check mark is present; when the filter is off, the check mark is absent. To remove the filter, choose the same option again.

Enabling and Disabling Scrolling

By default, the Message Log window is a continuously scrolling window. In other words, the text displayed in the window is adjusted with addition of each new message so that the most recent messages are always visible. However, there is a Scroll Lock option on the View pulldown menu that will turn off the scrolling feature. Choosing this option will lock the list of messages at the current position until the lock is removed. Alternatively, double clicking on a particular message will lock the list of messages at that message until scroll lock is disabled. The user will continue to have the option of manually scrolling through the list regardless of the status of the scroll lock option. This option operates as a toggle; the state of the scroll lock is indicated by a check mark ✔ adjacent to the menu option text. When the scroll lock is on, the check mark is present; when the scroll lock is off, the check mark is absent. To remove the scroll lock, choose the same option again.

High Level Status Windows

The Master provides access to several levels of status data for resources it monitors and controls. The Tracking Station Resources window is just one form the presentation of status data may take and it is intended to only present the very highest level status. For more detail, the user can activate and monitor high-level status windows. These windows present a high-level status summary of each unit of a particular resource type. See **Figure 6-6**[9].

Figure 6-6 A High Level Status Window

Opening a High Level Status Window

There are two ways to activate a high-level status window for a resource group. The user can choose from the list of resource types under the Window pulldown menu on the Monitor and Control process main menu. Or, the window can be activated by clicking the corresponding button on the Tracking Station

---

[8] Log files can be viewed with any ASCII text viewer.
[9] The window in Figure 6-6 is a high level window for the Metrum BVLDS digital cassette recorder.

Resources window. Once a high-level status window is activated, the contents of the window are updated on a regular basis as status is reported by the Remote Nodes.

The exact format and contents of the high-level status windows is resource type specific. However, each window includes a header at the top of the list, immediately below the menu bar. Also, the high level status is usually contained on one line of text. The unit numbers, OK/Error status flag, and the Switch Name are included on the display.

To close a high level status window, click the close button in the upper right hand corner of the window or choose **Close** from the high level status window menu.

Getting More Details (or, Starting the Device Interface)

The greatest amount of status details and control options are provided through device specific interface processes. The device interface processes provide front-panel like monitoring and controlling of tracking station resources. With a device interface, a user can monitor a thorough collection of parameters defining the status of the device[10]. In addition, a user can exercise full control of a device using the device interface application[11]. There are two ways to launch a device interface for a specific unit in remote control and monitor mode through interaction with the device high-level status window. One way is to double click on the unit to be controlled and monitored. The other way is to highlight the unit to be controlled and monitored and the choose the More Details option from the high level status window menu.

Saving Window Sizes and Positions

The size and positions of the Monitor and Control process windows can be saved by choosing the Save Window Positions Now option from the Window pulldown menu. Then, the next time the Monitor and Control process is started, the windows will be displayed with the saved sizes and positions.

Starting an Editor

The user has access to the Station Assets Editor, Profile Editor, Shipping Report Writer, and the device interfaces from the Monitor and Control main menu. These options are provide a mechanism with which the user can manipulate configuration files while continuing to monitor and control tracking station resource status.

The Profile Editor can be launched by selecting the Edit option on the Profile pulldown menu. Starting the Profile Editor in this manner enables the user to create or modify profiles not related to a currently scheduled support.

The Station Assets Editor and the Shipping Report Writer can be opened by selecting the appropriate option from the Tools pulldown menu.

Device configuration editors are opened by selecting the Device configuration editor option on the Tools menu and then selecting the desired device name. Starting the device interfaces this way enables the user to edit device configuration files for test scenarios. Device interfaces started in this manner do not enable the user to control or monitor a device.

Starting and Stopping System Processes

Ideally, the automation software is responsible for starting and stopping Master subsystem processes automatically. However, since situations may occur that may make it necessary to stop or restart a subsystem process, options to accomplish this are included in the System pulldown menu. The options in

---

[10] Hardware engineers and tracking station operators identified the collections of parameters defining the status of automated tracking station devices.
[11] Full control is defined as control of those features relevant to the device's operation in an automated tracking station.

this menu enable an authorized user to start or stop a specific subsystem process. The options themselves reflect the current status of each of the processes. In other words, if a process is not active, the System pulldown menu will contain an option to start it; or, if a process is active, the menu will contain an option to stop it. When the user chooses an option to start a process, the process will be started immediately. When the user chooses an option to stop a process, a warning will be displayed and a password-protected confirmation will be required before the process will be terminated.

Chapter 7 Using the Device Interfaces

For more information about using the device interfaces see **Appendix D: General Device Interface Certification Procedure**.

Local Edit Mode

In this mode, the GUI provides the user with a mechanism for manipulating Remote Node resident test configuration files. No control of the device is provided. Files created in this mode are not unit specific. The Load New and Apply Now buttons are disabled. The operator can only load a file using the open folder on the toolbar. Once a change has been made to the configuration file the file can be saved by using the toolbar or by clicking save on the drop down menu. The application is initialized to reflect documented configuration defaults.

A device GUI application is started in Local Edit Mode when the user double-clicks on the appropriate desktop shortcut icon at a Remote Node.

Local Control and Monitor Mode

Local control and monitoring of a device can be performed from the Remote Node to which the device is connected. It allows the user to review the current configuration of the device and make changes to that configuration as desired. Changes to configuration parameters are forwarded immediately to the device. The application is initialized to reflect the current configuration of the device.

Local control and monitoring of a specific device is initiated when the user double clicks on the appropriate desktop shortcut icon at a Remote Node.

Remote Edit Mode

In this mode, the GUI provides users with a tool for manipulating profile related and test configuration files. No control of a specific device is provided. Files created in this mode are not unit specific. The Load New and Apply Now buttons are disabled. The operator can only load a file using the open folder on the toolbar. Once a change has been made to the configuration file the file can be saved by using the toolbar or by clicking save on the drop down menu. The application is initialized to reflect documented configuration defaults or previously saved values.

Remote editing of a configuration file is initiated when the user clicks the "Edit Configuration" button on the Profile Editor Resource Requirements window or when the user double clicks on the appropriate desktop shortcut icon at a Master.

Remote Control and Monitor Mode

Remote controlling and monitoring of a device can be performed from the Master. This mode of operation is only available during a support, between setup and takedown of that support. It allows the user to review the current configuration of the device and make changes to that configuration as desired. Changes to configuration parameters are forwarded immediately to the device. The application is initialized to reflect the current configuration of the device.

Remote control of a device is initiated when the user double clicks on the appropriate high level status for the unit as displayed in the Master Monitor and Control device status window. Any Master can remotely control a device during a support.

Appendix A: Glossary

.bsz
The extension attached to configuration files for the Decom Systems 7715 bit synchronizer.

.dat
The extension attached to data files.

.dmd
The extension attached to configuration files for the Aydin 329A BPSK demodulator.

.dtj
The extension attached to configuration files for the Datatape 3700J analog tape recorder.

.DTP
The extension attached to configuration files for the Avtec 1000 PTP.

.fsz
The extension attached to configuration files for the General Data Products 225D frame synchronizer.

.hps
The extension attached to configuration files for the HP 3325B synthesizer.

.khf
The extension attached to configuration files for the Krohn
Hite 3905B programmable filter.

.mdu
The extension attached to configuration files for the General Data Products 783 M BPSK
subcarrier/modulator .

.**rfm**
The extension attached to configuration files for the HP E1366/67A RF matrix switch.

.sim
The extension attached to configuration files for the General Data Products 233 PCM simulator.

.tdf
The extension attached to configuration files for the WFF TDF.

.vld
The extension attached to configuration files for the Metrum BVLDS cassette digital tape recorder.

.xxx
The extension attached to matrix files for the PCM Simulator.

11 meter Interface process
A Master subsystem process that coordinates the communication between the Scientific Atlanta 11 meter
Antenna and the Master.

A
AGS
The Alaska Ground Station, located at the University of Alaska, Poker Flat Research Range near Fairbanks,
Alaska, USA.

AOS
Acquisition of signal (satellite)

About box
An application's About Box contains information describing the latest version of the application.

accelerator key
A keystroke or combination of keystrokes that invokes a particular command.

active toolbar
The toolbar appropriate to the current mode of operation of an application. Before a file is open the standard toolbar is displayed; after a file is open the edit mode toolbar is displayed.

arrow key
The up, down, left, and right arrow keys. These keys can be used to move the cursor through a list of options.

automated schedule
A subset of the operational schedule that includes only those events scheduled for an antenna that has been included in the automation effort.

AWOTS
The Automated Wallops Orbital Tracking Station at the NASA/GSFC/Wallops Flight Facility in Wallops Island, Virginia, USA.

B
BPSK
Binary phase shift key.

block diagram
A graphical representation of high level resource status and the data for all tracking station resources.

busy cursor
A cursor that assumes an hourglass shape to indicate that the application is busy and temporarily not accepting input.

button control
A graphical control operated by a user to issue a command to an application.

BVLDS
Buffered very large data storage.

C
CCSDS
 Consultative Committee for Space Data Systems

cascade
To arrange an application's windows within the application client area such that the title bar of every window is visible and the windows are overlapped.

check box
A square box operated by the user to choose from a set of related but independent options. A checked check box contains an X. Repeatedly checking a check box toggles it from state to state. A standard check box has two states: checked and unchecked.

D

desktop
The visible screen area.

device hardware specifications
Any specific information about a device including environmental limitations, physical dimensions and dip switch settings. This information is accessed from a device GUI application and is displayed in a text viewer.

dialog box
A window used to solicit input from the user.

disabled
A window that can not receive input from the keyboard, mouse, or other input device.

double-click
To press and release a mouse button twice in quick succession.

drag
To move the mouse while holding down a mouse button.

drop list
A graphical control that presents a list of selectable items when the user selects the drop down arrow next to the control. The list of selectable items can not be edited by the user.

E

edit field
A graphical control that allows the user to enter and edit text.

edit mode toolbar
The toolbar that performs some of the functions of an application when a file is open.

F

FTP
File transfer protocol

G

GDP
General Data Products

General Resources Manager
The process that coordinates communication between the Master and the Nodes.

GRM
The General Resources Manager.

GSFC
Goddard Space Flight Center

GUI
A Graphical User Interface is a process that gathers input from and presents output to a user.

H

highlight
To select a portion of text by positioning the mouse over a text field and moving the mouse while holding down the mouse button.

HP
Hewlett Packard

I

IP
internet protocol

icon
A small bitmap that usually represents a minimized application. Icons may also serve as symbols in warning messages or other windows.

indicator light
A small bitmap that usually represents a status from a device.

J

K

L

LOS
Loss of signal (satellite).

list box
A graphical control window that contains a list of items that can be selected by the user.

local control

M

M&C
The Monitor and Control application.

MDDF
Minimum delay data format

MGS
The McMurdo Ground Station in McMurdo, Antarctica.

Master
The workstation on which has been installed a collection of processes that provide centralized, automated, remote controlling and monitoring of tracking station assets.

Master password
A subsystem password is used to restrict access to some functions that may interfere with the Master's ability to accomplish scheduled support activity.

Master Password Editor
A utility that can be used to modify the Master password.

maximize
To enlarge a window to the size of the screen or the containing window.

maximize box
A window component that, when clicked, enlarges a window to the size of the screen or the containing window.

menu bar
A list of items the represent an application's commands. The menu bar typically contains one or more menu items can select to display pulldown menus. It is located at the top of the application window, immediately below the title bar.

minimize
To reduce a window to an icon.

minimize box
A window component that, when clicked, reduces a window to the size of an icon.

minimized window
A window that has been reduced to an icon.

Monitor and Control process
The Master subsystem process that coordinates the controlling and monitoring of tracking station resources. This is the primary user interface to automated tracking station resources.

N

NASA
National Aeronautics and Space Administration

NASCOM
National Aeronautics and Space Administration Communications

Node
A workstation on which has been installed a collection of processes that provide controlling and monitoring of a subset of tracking station assets.

non-automated schedule
A subset of the operational schedule that includes events scheduled for an antenna that has not been included in the automation effort.

O

operational schedule
The schedule delivered to the tracking station Master from the scheduling office that includes automated and non-automated support requirements.

P

PC
Personal computer

PCM
Pulse code modulation

PSK

Phase shift key

PTP
Programmable telemetry processor

process
An executing application that consists of a private virtual address space, code, data, and other operating system resources, such as files, pipes, and synchronization objects that are visible to the process. A process also contains one or more threads that run in the context of the process.

profile
A collection of files defining the configuration of tracking station resources for an anticipated support.

Profile Editor
A Master subsystem process used to define the configuration of tracking station resources for an anticipated support.

pulldown menu
A menu that is displayed when the user selects a particular entry from a menu bar.

Q

R

radio button
A round button operated by the user to choose from a set of related but mutually exclusive options. The check mark for a radio button is a black dot. A radio button can assume two states: checked (dot) or unchecked (no dot).

read-only text
A field that contains text presented to the user for the user's information and accepts no user input.

ready cursor
A cursor that assumes an arrow or blinking line shape to indicate that the application is ready to accept input from the keyboard or the mouse.

remote control

Remote Node

resize
To alter the size of a window by stretching or shrinking the window border.

S

SA
Scientific Atlanta

SSB
The Station Status Broadcaster application.

SSD
The Station Status Display application.

SGS
The Svalbard Ground Station in Longyearbyen, Svalbard, Norway.

Scheduler process
The Scheduler process is the Master subsystem process responsible for the ingestion and interpretation of the operational schedule.

scroll bar
A sliding control that a user can manipulate to shift the contents of a window's client area either horizontally or vertically. It is usually positioned along the left and/or bottom window border when the window's client area is greater than the size of the displayed window.

set focus
To change the keyboard focus to the specified window.

shortcut key
Some menu commands can be executed by pressing defined shortcut keys. For example, in most applications, pressing the F1 key starts on
line Help and Ctrl + S saves the open file.

spin control
A control including a pair of arrows that the user can click to adjust a value displayed in an adjacent, edit field window.

standard toolbar
The toolbar that performs some of the functions of an application before a file is open.

static text
A label or text control that accepts no user input.

Station Assets Editor
The Master subsystem process used to define station specific characteristics required for proper execution of the Master subsystem software.

status bar
A control bar that contains a row of text output panes that display status indicators. It is typically positioned along the bottom of a window.

T

TDF
Tracking data formatter

TDM
Time division multiplexed

TOTS
S-Band Transportable Orbital Tracking System used for orbital and range support.

TR code
This is a code used by tracking station operators to identify the support configuration.

TTL

tile
To arrange an application's windows within the application client area such that all windows are the same size and adjacent to each other. The windows will be arranged so that the client area is filled but no windows overlap.

title bar
The top border of a window containing a title defining the application or contents associated with the window.

tool tip
A tiny pop-up window that presents a short description of a toolbar button's action. Tool tips are displayed when a user positions the mouse cursor over a button for a period of time.

toolbar
A control bar that contains a row of pushbuttons and/or radio buttons. It is typically positioned along the top of a window, immediately below the main menu. It can be displayed or hidden by selecting the Toolbar option from the View menu.

U
UDP
Unreliable data packet

UTDF
Universal tracking data format

V

W
WFEP
Wallops front end processor

WOTIS
The Wallops Orbital Tracking Information System (???).

WOTIS Interface process
The Master subsystem process that anticipates and accepts the operational schedule from the Wallops Orbital Tracking Information System.

WPS (or WGS)
The Wallops Ground Station, located at the NASA/GSFC/Wallops Flight Facility in Wallops Island, Virginia, USA.

window
A rectangular area of the screen bounded by a border.

window control menu
A menu that include options to move or resize a window, or switch to another window. The menu is obtained by positioning the cursor over the window icon in the upper left
hand corner and clicking the left mouse button.

X

Y

Z

Automated Tracking Station User's Manual

Appendix B: WOTIS – Master Communication

B.1 Introduction

This document defines the method of communication of the operational schedule from the Wallops Orbital Tracking Information System (WOTIS) to the automated tracking station Master. Specifically, it describes the procedure by which the Master is notified of a new operational schedule as well as the contents and format of the file delivered to the Master.

This document supersedes any previous documentation defining the communication of the operational schedule from WOTIS to the Master.

B.2 Delivery and Notification of a New Schedule

The Master receives scheduling information via Ethernet. The operational schedule is sent to each Master via FTP. A file containing new schedule information is transferred to the following destination on the Master: **c:\Master\Schedule\wotrs.mas**.

Each Master is notified that new schedule data has been delivered by a message sent from WOTIS to a socket (**22375**) created and maintained by the Master. The Master does not begin ingesting the new schedule until this notification is received.

B.3 Format and Contents of the Operational Schedule

The file delivered from WOTIS to the Master is an ASCII text file. The format of the file is as follows:
- ❑ Security Code
  The first line of the file is a security code that signifies to the Master that the schedule file is a valid operational schedule file. This code includes up to eight characters.
- ❑ Day of Year and Year
  The next line of the file includes the day of year and year (including century) to which the schedule pertains.
- ❑ Event Information for Each Event
  A complete description of each event to be conducted by the tracking station is included. Event information is assumed to be chronologically ordered and conflict free. For a detailed description of the information required for each event refer to section **4.0 Format and Contents of Event Information**.
- ❑ Terminator
  The last line of the file, immediately following the last element of the last event, is the end of schedule terminator **###**.

B.4 Format and Contents of Event Information

The section describes the information included for each scheduled event. The definition, format, and default is provided for each element. If an element is not available for or not relevant to an event, the default value is included. (Note: if the default value for an element is defined as Not applicable, that element of the event is essential and must be included for all events.)

- ❏ Satellite ID[*]
  - Definition: The satellite for which tracking station support is scheduled.
  - Format: Up to eight alphanumeric characters.
  - Default: Not applicable.

- ❏ Site ID
  - Definition: An identifier of the station scheduled to perform the event.
  - Format: A character string containing one of the following site IDs:
    AGS
    MGS
    SGS
    WPS
  - Default: Not applicable.

- ❏ WOTRS Line Item
  - Definition: A WOTRS event specific identifier.
  - Format: Up to ten alphanumeric characters.
  - Default: Not applicable.

- ❏ Operation[*]
  - Definition: The operation to be performed.
  - Format: A single integer corresponding to one of the following operations:
    0   None (NON)
    1   Scheduled Support (SUP)
    2   Scheduled Playback (PBK)
    3   Scheduled Test (TES)
    4   Time Window Playback (TWP)
    5   Down (DWN)
  - Default: 0

---

[*] Fields marked with this symbol are used by the Master in determining the appropriateness of a support profile for a scheduled event.

Automated Tracking Station User's Manual

- □ Receive and Transmit Antenna Specifications[*]
  Definition: Receive and transmit antenna identifiers scheduled for the support.
  Format: Two integers, separated by a space, and corresponding to the following codes:
                 None
                 9 meter
                 7.3 meter #1
                 7.3 meter #2
                 Satan 1
                 Satan 2
                 UHF – Satan 3
                 ADAS
                 Meteosat
                 6 meter
                 SA 10 meter
                 SA 11 meter
                 Scamp
                 TOTS #1
                 TOTS #2
                 TOTS #3
                 LEO-T 5 meter
                 26 meter
                 15 meter
                 34 meter
  Default: 0 0

- □ AOS and LOS Date
  Definition: The dates of Acquisition of Satellite (AOS) and Loss of Satellite (LOS).
  Format: AOS day of month, month, year (including century), and LOS day of month, month, year (including century); where all fields are integers and separated by a space.
  Default: Not applicable.

- □ AOS and LOS Time
  Definition: The times of AOS and LOS.
  Format: AOS hours, minutes, whole seconds, and LOS hours, minutes, whole seconds; where all fields are integers and AOS and LOS are separated by a space.
  Default: Not applicable.

- ❑ Event Start and Stop Date
  Definition: The start and stop dates for the event.
  Format: Start day of month, month, year (including century), and Stop day of month, month, year (including century); where all fields are integers and separated by a space.
  Default: Not applicable.

- ❑ Event Start and Stop Time
  Definition: The start and stop dates for the event.
  Format: Start hours, minutes, whole seconds, and Stop hours, minutes, whole seconds; where all fields are integers and Start and Stop are separated by a space.
  Default: Not applicable.

- ❑ Ranging
  Definition: A flag indicating if ranging is to be performed in conjunction with the event.
  Format: An integer, where 0 corresponds to false and 1 corresponds to true.
  Default: 0

- ❑ Support Type
  Definition: The type of support with which the event is associated.
  Format: Four flags, separated by a space. Each flag indicates True (1) or False (0) for each of the following support types: Receive Link, Commanding, Tracking, Real-time Data.
  Default: 0 0 0 0

- ❑ Data Line
  Definition: The data line scheduled to be used during the support, specified in KB.
  Format: An integer.
  Default: 0

- ❑ Data Rate
  Definition: The data rate and format describing the spacecraft downlink.
  Format: Up to 5 characters, A-G.
  Default: *

- ❑ Doppler Specification
  Definition: An indication of the RF link to be used for the event.
  Format: A single integer corresponding to one of the following values:

	-1	None
	0	Full two-way Doppler
	1	One-way Doppler
	2	Two-way Doppler
	3	Three-way Doppler
Default:	-1	

❑ Voice Period
Definition: The start and stop times for the voice period.
Format: Start hours and minutes and Stop hours and minutes, where all fields are integers and start and stop times are separated by a space.
Default: 0000 0000

❑ Orbit Number
Definition: The satellite orbit number.
Format: An integer.
Default: 0

❑ Tape ID
Definition: An identifier for a tape to be used for a playback.
Format: Up to 15 alphanumeric characters including a three character site ID, a two character recorder ID (AX = Ampex, MM = Metrum BVLDS), and a 10 character tape volume label.
Default: NONE

❑ Start Block Address
Definition: The start block address for a playback.
Format: Up to 10 alphanumeric characters.
Default: NONE

❑ End Block Address
Definition: The end block address for a playback.
Format: Up to 10 alphanumeric characters.
Default: NONE

❑ Band
Definition: The downlink band for the antenna.
Format: Two alphanumeric characters.
Default: XX

- Ephemeris
  Definition: The ephemeris data for the event.
  Format: An ASCII text string including a format specification followed by data. The format specification (and the corresponding data) must be one of the following: Brouwer, tod, or iirv. The data/time must be in a modified Julian format.
  Default: NOEPHEMERIS

- Remarks
  Definition: Remarks to be relayed to the operator.
  Format: Up to eighty alphanumeric characters.
  Default: NONE

Automated Tracking Station User's Manual

Appendix C: Pass Results Format

The Scheduling and Master Teams met on Tuesday, 9/23, to work out the final format of the Pass Results File which will be generated upon takedown after an activity at any WFF tracking station controlled with the Automated Orbital Tracking System. The name of the file will be PRF[Station Designator], where Station Designator is: W for WGS, M for MGS, S for SGS, and A for AGS. As soon as takedown is complete the file is to be generated and sent via FTP to the WISAC component of WOTIS, into the WOTRS mailbox (machine IP address, user name, password, and directory are supplied separately from this memo).

The current PRF[ ] file has been changed in the following respects:

1. Five new fields have been added for data quality generated by a Programmable Telemetry Processor (PTP). This should cover both the AVTEC PTP and the ADEOS-II Data Stripper.. The Master may need to summarize the information in the log files from these instruments to provide these parameters.
2. A new field has been added for tape recorder data quality, where provided by a recorder. Again, the Master may need to perform a simple calculation to generate a percent from raw log values.
3. The fields Tracked & Recorded have been transposed.
4. Tracked will be given a value of Y for Yes if the combined movement of the antenna (Azimuth & Elevation) from the beginning to the end of the track exceeds 10 degrees. Otherwise, tracking is assumed to have been unsuccessful, and Tracked = N for No.
5. Recorded will be given a value of Y for Yes if, where the Event Designator is SUP or SPS (indicating a S/C support is being performed), the end address minus the start address for the tape recorder exceeds 200. Otherwise, it will be assumed that the tape did not move enough for recording to be successful, and Recorded will be N for No.

As before, if a field has no value, it should be blank filled with the proper number of blanks (so proper field alignment is always maintained).

Resulting from a better definition of PTP capabilities (reference WOTIS note, "PTP Data Quality Requirements", dated 1/26/98), a sixth single character field has been added to #1 above to define the PTP operation being reported: Command Processing (C), non-CCSDS Telemetry Processing (T), CCSDS Telemetry Processing (R). In addition, the items in #1 have been reordered.

The new format is given below.

PRF[ ] File Format

PRF[ ] File Format (continued)

Telemetry Processor	TP#1	TP#2	TP#3	TP#4	TP#5	TP Op
AVTEC PTP Command	# Frames Xmitted	# Headers received	# Headers accepted	# Data Units sent	# Command echos	PTP Stream ID
AVTEC PTP Telemetry	# Bytes output to file or socket	Total # of frames received	# CRC errors (CCSDS)	# correctable Reed-Soloman frames (CCSDS)	# un-correctable Reed-Soloman frames (CCSDS)	PTP Stream ID
TSI Data Stripper (ADEOS S/C)	Total number of frames processed	Number of corrected frames	Number of uncorrected frames	Back_to_Search count	Bit Error Rate	Downlink Band (X1|X2|X3)

Appendix D: General Device Interface Certification Procedure

 (Local Mode Only)

D.1 Scope

This test plan verifies all of the general screen-based portions of the automated tracking station graphical user interface (GUI) software. This test plan addresses only the local modes, therefore the GUI is to be tested in only the following control modes: Local Edit and Local Control/Monitor.

D.2 Environment

• Pentium processor or better
• Minimum 16 megabytes RAM
• Windows NT 4.0
• Hardware device to be tested

D.3 Resources

The following files need to reside in the same directory:

• Application executable (*Device*S*omething***.exe**)
• Initialization file defined by APP_INIT_FILE in *Device*S*omething***. h** (*Device*S*omething***.ini**)
• Device configuration file defined by APP_CONFIG_FILE in *Device*S*omething***. h**
(*Device*S*omething.something*)
• Default device configuration file defined by APP_DEFAULT_FILE in *Device*S*omething***. h**
(*Device*S*omething*Default.*something*)
• Informational file defined by APP_UPDATE_FILE in *Device*S*omething***. h** (*Device*S*omething***.txt**)
• File defined by DEVICE_HARDWARE_SPECS_FILE in *Device*S*omething***. h**
(*Something***HardwareSpecs.txt**)
• Device status update file **WcDev***DeviceMakeModel***.status**
• Application help file (*Device*S*omething***.hlp**)

The following files are also required:

• Support dynamic link libraries (*DeviceMakeModel***.dll**)
• Applicable MFC dll's (i.e. **mfc42u.dll**), MSVC dll's (i.e. **msvcrt40.dll**)

D.4 Assumptions

• The tester is knowledgeable in the basics of Windows program operation including features such as pull-down menus, toolbars, and "drag and drop".
• The GUI has been installed on the test computer.
• The device to be tested has been setup and correctly connected to the computer.

D.5 Procedures

All menu items and toolbars will be tested in each applicable mode. Response of the device in Local/Control and Remote Control modes will be verified to the extent possible for the particular device. Any data calculations made, as part of the data entry, will be validated.

A checklist will be provided for every item of each mode to facilitate testing. Check the box for each item that passes. If any item fails, add a comment to the Comments Table at the end of this document. The comment should cross reference the item number that failed and give details about why the item failed the test.

The tester may also add any other comments to the comments table. If the comments pertain to a particular item then cross reference with the item number.

Checklist

A check box precedes each of the following procedures. This check box is to be checked if and only if the test passes; i.e. the statement is true.

D.6.1.0 <u>Local Edit</u>

❑   **1.1.0** The application initializes by double clicking on appropriate icon without any errors.   No command line arguments expected.

❑   **1.2.0** The following menu options are available before a file is open: **File**, **View** and **Help**.

❑   **1.2.1** The following **File** pull-down menu options are available: **New**, **Open**, **Print Setup** and **Exit**.

❑   **1.2.1.1** Choose **Print Setup**. Visually inspect print options in the standard *Microsoft* **Print Setup** window.

❑   **1.2.1.1.1** Click on **Cancel**.

❑   **1.2.2** The following **View** pull-down menu options are available: **Toolbar** and **Status Bar**.

❑   **1.2.2.1** Choose **Toolbar** (which is checked). Hides toolbar:

Toolbar

❑   **1.2.2.1.1** Choose **Toolbar** again. Displays toolbar.

❑   **1.2.2.2** Choose **Status Bar** (which is checked). Hides status bar:

Status Bar

❑   **1.2.2.2.1** Choose **Status Bar** again. Displays status bar.

❑   **1.2.3** The following **Help** pull-down menu options are available: **Contents**, **Using Help**, **About…** and **Device Hardware Specifications**.

❑   **1.2.3.1** Choose **Contents**. Opens device specific on-line help file.

❑   **1.2.3.1.1** Choose **Exit** from the **File** menu or standard *Microsoft* close (**X**) button in the upper right-hand corner of the *Help* window.

❑   **1.2.3.2** Choose **Using Help**. Opens general *Microsoft* on-line help file.

❑   **1.2.3.2.1** Click on **Cancel** or standard *Microsoft* close (**X**) button in the upper right-hand corner of the *Help Topics: Windows Help* window.

❑   **1.2.3.3** Choose **About**. Displays window containing application icon, application name, version number and last modification date.

❑   **1.2.3.3.1** Choose **What's New**. Displays text file in *Notepad*. Text file describes application and its recent modifications.

❑   **1.2.3.3.1.1** Choose **Exit** from the **File** menu or standard *Microsoft* close (**X**) button in the upper right-hand corner of the *Notepad* window.

❑ **1.2.3.4** Choose **Device Hardware Specifications**. Displays informational text file in *Notepad*.

    ❑ **1.2.3.4.1** Choose **Exit** from the **File** menu or standard *Microsoft* close (**X**) button in the upper right-hand corner of the *Notepad* window.

❑ **1.3.0** The following tool buttons are available: **New**, **Open**, **Print Setup**, **Toggle ToolBar**, **Toggle StatusBar**, **About**, **Device Specifications** (the last button below is an example).

<div align="center">

Tool buttons

</div>

❑ **1.3.1** Each tool button has appropriate Tool Tip displayed when cursor is positioned over button.

❑ **1.3.2** Click on **Print Setup**. Visually inspect print options in the standard *Microsoft* Print Setup window.

<div align="center">

Print Setuptool button

</div>

    ❑ **1.3.2.1** Click on **Cancel**.

❑ **1.3.3 Toggle ToolBar** – Click on ToolBar button. Hides toolbar.

<div align="center">

Toggle ToolBar tool button

</div>

    ❑ **1.3.3.1** Choose **Toolbar** from the **View** menu. Displays toolbar.

❑ **1.3.4 Toggle StatusBar** – Click on StatusBar button. Hides status bar.

<div align="center">

Toggle StatusBar tool button

</div>

    ❑ **1.3.4.1** Choose **StatusBar** from the **View** menu. Displays status bar.

❑ **1.3.5** Click on **About**. Displays window containing application icon, application name, version number and last modification date.

<div align="center">

About tool button

</div>

❑ **1.3.6** Click on **Device Specifications**. Displays informational text file in *Notepad*. (The button below is an example.)

<div align="center">

Device Specifications tool button

</div>

1.4.0 Open a new file without error by choosing the following menu option:

❑ **1.4.1 New** – A set of windows displays containing default parameters for the device.

    ❑ **1.4.1.1** When the standard *Microsoft* close (**X**) button in upper right-hand corner of each of these windows is clicked, the window does not close, but is minimized.

    ❑ **1.4.1.2** For each window closed in the above step, restore it to the open state by double-clicking on the title bar or clicking on the first of three buttons on the right-hand side of the title bar of the minimized window.

    ❑ **1.4.1.3** Move a few windows by right-clicking on the title bar of the window holding the mouse down and dragging it. Do not restore window position.

❑   **1.4.1.4** Resize a few windows by positioning the cursor over an edge of the window, right–clicking on the mouse, holding it down and dragging it. Do not restore window size.

❑   The following **File** options are now among those available after a file is open: **Print** and **Print Preview.**

❑   **1.4.2** Choose **Print Preview**. Displays current GUI parameters/settings in a standard text viewer.

❑   **1.4.2.1** Click on **Close**.

❑   **1.4.3** Choose **Print**. Displays the standard *Microsoft* **Print** window, which allows user to make print choices.

❑   **1.4.3.1** Click on **OK.** Prints parameters currently displayed in GUI.

❑   **1.5.0** The following **Window** menu options are available after a file is open: **Cascade, Tile, Arrange Icons,** Window #1 (i.e. **Settings**), Window #2, #3, etc. (i.e. **Control, Status**). The number of "Windows" should coincide with the number of windows open not including the main window of the application itself.

Note: Minimize the "bottom" window for the next two steps.

❑   **1.5.1** Choose **Cascade** – Arranges windows not minimized in cascading pattern within GUI.

❑   **1.5.2** Choose **Tile** – Arranges windows not minimized in tiled pattern within GUI.

Note: Minimize another window for the next step. Move the minimized windows to different locations by right-clicking on the title bars, holding the mouse down and dragging each.

❑   **1.5.3** Choose **Arrange Icons** – Arranges windows minimized on bottom of GUI.

Note: Open all minimized windows and choose **Cascade**.

❑   **1.5.4** Choose Window #1 (i.e. **Settings**), if applicable: _____ (write window name) – Brings window #1 to foreground or "on top" of others.

❑   **1.5.5** Choose Window #2 (i.e. **Control**), if applicable: _____ (write window name) – Brings window #2 to foreground or "on top" of others.

❑   **1.5.6** Choose Window #3 (i.e. **Status**), if applicable: _____ (write window name) – Brings window #3 to foreground or "on top" of others.

❑   **1.5.7** Choose Window #4, if applicable: _____ (write window name) – Brings window #4 to foreground or "on top" of others.

❑   **1.5.8** Choose Window #5, if applicable: _____ (write window name) – Brings window #5 to foreground or "on top" of others.

❑   **1.5.9** Choose Window #6, if applicable: _____ (write window name) – Brings window #6 to foreground or "on top" of others.

❑   **1.5.10** Choose Window #7, if applicable: _____ (write window name) – Brings window #7 to foreground or "on top" of others.

1.6.0 Specific controls – For *specific* functionality that should be provided in a particular GUI, see the requirements document for the specific device being controlled. In some cases, controls are dependent on other controls; i.e. controls may not be enabled until another control is chosen. Again, see the requirements document. The following are the *possible* controls available, and *in general*, should operate in the following manner by clicking the mouse on them:

Note: You may need to maximize each window and/or the main application window or use the scroll bar(s) in order to view all of the controls. Choose the second (middle) button in the right-hand side of the title bar of the window.

❑ **1.6.1 Apply Now** (if applicable) – In the **Settings, Control** or similar window(s) this pushbutton is disabled/grayed.

<p align="center">Disabled Apply Now button</p>

❑ **1.6.1a Load New** (if applicable) – In the **Control** or similar window(s) this pushbutton is disabled/grayed.

<p align="center">Disabled Load New Button</p>

❑ **1.6.2 Reset** (if applicable) – In the **Settings, Control** or similar window(s) this pushbutton is disabled/grayed.

<p align="center">Disabled Reset button</p>

❑ **1.6.3** Switch (if applicable) – In the **Settings, Control** or similar window(s) this button is disabled/grayed:

<p align="center">Disabled Switch</p>

❑ **1.6.4** Edit box (if applicable) – In the **Settings, Control** or similar window(s) this control is
❑ enabled unless it is dependent on another selection.
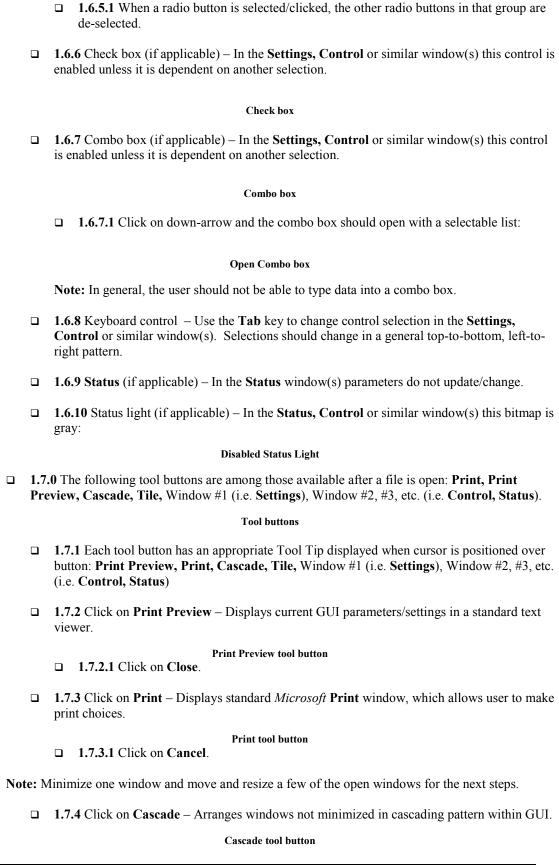
<p align="center">Edit box</p>

❑ **1.6.4.1** Spin (Up/down) control (if applicable) – Used with some edit boxes, this control is enabled with its associated edit box.

<p align="center">Spin control</p>

❑ **1.6.4.1.1** When the up-arrow portion of the spin control is clicked the parameter in the associated edit box increments. To change the digit incrementing, highlight the digit and click the left mouse button.

❑ **1.6.4.1.2** When the down-arrow portion of the spin control is clicked the parameter in the associated edit box decrements. To change the digit decrementing, highlight the digit and click the left mouse button.

❑ **1.6.5.** Radio buttons (if applicable) – In the **Settings, Control** or similar window(s) these controls are enabled unless it is dependent on another selection.

<p align="center">Radio button group</p>

❑ **1.6.5.1** When a radio button is selected/clicked, the other radio buttons in that group are de-selected.

❑ **1.6.6** Check box (if applicable) – In the **Settings, Control** or similar window(s) this control is enabled unless it is dependent on another selection.

Check box

❑ **1.6.7** Combo box (if applicable) – In the **Settings, Control** or similar window(s) this control is enabled unless it is dependent on another selection.

Combo box

❑ **1.6.7.1** Click on down-arrow and the combo box should open with a selectable list:

Open Combo box

Note: In general, the user should not be able to type data into a combo box.

❑ **1.6.8** Keyboard control – Use the **Tab** key to change control selection in the **Settings, Control** or similar window(s). Selections should change in a general top-to-bottom, left-to-right pattern.

❑ **1.6.9** Status (if applicable) – In the **Status** window(s) parameters do not update/change.

❑ **1.6.10** Status light (if applicable) – In the **Status, Control** or similar window(s) this bitmap is gray:

Disabled Status Light

❑ **1.7.0** The following tool buttons are among those available after a file is open: **Print, Print Preview, Cascade, Tile,** Window #1 (i.e. **Settings**), Window #2, #3, etc. (i.e. **Control, Status**).

Tool buttons

❑ **1.7.1** Each tool button has an appropriate Tool Tip displayed when cursor is positioned over button: **Print Preview, Print, Cascade, Tile,** Window #1 (i.e. **Settings**), Window #2, #3, etc. (i.e. **Control, Status**)

❑ **1.7.2** Click on **Print Preview** – Displays current GUI parameters/settings in a standard text viewer.

Print Preview tool button

❑ **1.7.2.1** Click on **Close**.

❑ **1.7.3** Click on **Print** – Displays standard *Microsoft* **Print** window, which allows user to make print choices.

Print tool button

❑ **1.7.3.1** Click on **Cancel**.

Note: Minimize one window and move and resize a few of the open windows for the next steps.

❑ **1.7.4** Click on **Cascade** – Arranges windows not minimized in cascading pattern within GUI.

Cascade tool button

❑   **1.7.5** Click on **Tile** – Arranges windows not minimized in tiled pattern within GUI.

Tile tool button

Note: Open all minimized windows and choose **Cascade** for the next steps.

❑   **1.7.6** Click on Window #1 (i.e. **Settings**), if applicable: _____ (write window name) – Brings window #1 to foreground or "on top" of others.

Settings tool button

❑   **1.7.7** Click on Window #2 (i.e. **Control**), if applicable: _____ (write window name) – Brings window #2 to foreground or "on top" of others.

Control tool button

❑   **1.7.8** Click on Window #3 (i.e. **Status**), if applicable: _____ (write window name) – Brings window #3 to foreground or "on top" of others.

Status tool button

❑   **1.7.9** Click on Window #4, if applicable: _____ (write window name) – Brings window #4 to foreground or "on top" of others.

❑   **1.7.10** Click on Window #5, if applicable: _____ (write window name) – Brings window #5 to foreground or "on top" of others.

❑   **1.7.11** Click on Window #6, if applicable: _____ (write window name) – Brings window #6 to foreground or "on top" of others.

❑   **1.7.12** Click on Window #7, if applicable: _____ (write window name) – Brings window #7 to foreground or "on top" of others.

❑   **1.8.0** The following **File** options are among those available after a file is open: **Save** and **Save As.**

      ❑   **1.8.1 Save** – Make a change to one of the parameters in the **Settings** (or similar) window and make a note of that change below. Choose **Save**. User will be prompted to save file to a new name in a **SaveAs** dialog box since the file was opened as **New**.

         Parameter: _____   Changed to: _____

         ❑   **1.8.1.1** Click on **Cancel.**

      ❑   **1.8.2** Choose **Save As** – Save a new local configuration file by entering "**Test**" in dialog displayed. The file extension (i.e. ******.bsz*) will automatically be added to the name of the file.

         ❑   **1.8.2.1** Click on **Save**.

         ❑   **1.8.2.2** If message asks to overwrite an existing file, answer **Yes**.

❑   **1.9.0** The following **File** option is now among those available after a file is open: **Close**.

      ❑   **1.9.1** Choose **Close** – Closes open file. User may be asked to save open or new file (see above).

❑   **1.10.0** The following **File** option is now among those available: **Open.**

❑ **1.10.1** Choose **Open** – Opens local configuration file. Choose "**Test.xxx**" (i.e. Test.bsz) from window. You may have to scroll until the file is visible. The same set of windows displays as when a new file was open in step **1.4.1 (New)**.

  ❑ **1.10.1.1** Check parameter changed in step **1.8.1** (**Save**) and make sure it is what was saved.

❑ **1.11.0** The following tool buttons are now among those available: **Save** and **Save As.**

❑ **1.11.1 Save** – Make a change to another parameter (different from step **1.8.1**) in **Settings** (or similar) window and make a note of that change.

Parameter: _____   Changed to: _____

  ❑ **1.11.1.1** Click on **Save** tool button to save file.

Save tool button

    ❑ **1.11.1.1.1** If message asks to overwrite an existing file, answer **Yes**.

❑ **1.11.2** Click on **Open** tool button.

Open tool button

❑ **1.11.3** Repeat step **1.10.1.1**.

❑ **1.12.0** The following tool button is now among those available: **Close.**

Close tool button

❑ **1.12.1** Click on **Close** – Closes open file. User may be asked to save open or new file (see step **1.8.2.1**).

  ❑ **1.12.1.1** If message asks to overwrite an existing file, answer **Yes**.

❑ **1.13.0** The following menu option is available: **Exit**.

❑ **1.13.1** Choose **Exit** – Terminates application. User may be asked to save open or new file (see step **1.8.2.1**), answer **No**.

Note: At this point, the GUI has been completely checked in Local Edit.

D.7.2.0 <u>Local Control/Monitor</u>

Note: Before proceeding with this section, locate the device being controlled by the GUI and make sure it is on.

❑ **2.1.0** The application initializes by double clicking on appropriate icon without any errors. (The command line arguments are already provided in the icon's properties.)

❑ **2.2.0** The following menu options are available: **File**, **View, Window,** and **Help**.

❑ **2.2.1** The following **File** pull-down menu options are among those available: **Print Preview, Print, Print Setup** and **Exit**.

  ❑ **2.2.1.1 Print Preview** – Displays parameters currently in GUI in a text viewer.

       ❑   **2.2.1.1.1** Click on **Close.**

  ❑   **2.2.1.2 Print Setup** – Visually inspect print options in the standard *Microsoft* **Print Setup** window.

       ❑   **2.2.1.2.1** Click on **Cancel.**

  ❑   **2.2.1.3 Print** – Displays the standard *Microsoft* **Print** window, which allows user to make print choices.

       ❑   **2.2.1.3.1** Click on **OK** – Prints parameters currently displayed in GUI.

❑   **2.2.2** The following **View** pull-down menu options are available: **Toolbar** and **Status Bar**.

  ❑   **2.2.2.1 Toolbar** – Choose **Toolbar** (which is checked). Hides toolbar.

<center>**Toolbar**</center>

       ❑   **2.2.2.1.1** Choose **Toolbar** again. Displays toolbar.

  ❑   **2.2.2.2 Status Bar** – Choose **StatusBar** (which is checked). Hides status bar.

<center>**Status Bar**</center>

       ❑   **2.2.2.2.1** Choose **StatusBar** again. Displays status bar.

❑   **2.2.3** A set of windows is displayed containing parameters from the device. This can be verified by checking the front panel of the device with those in the GUI.

  ❑   **2.2.3.1** When the standard *Microsoft* close (**X**) button in upper right-hand corner of each of these windows is clicked, the window does not close, but is minimized.

  ❑   **2.2.3.2** For each window closed in the above step, restore it to the open state by double-clicking on the title bar or clicking on the first of three buttons on the right-hand side of the title bar of the minimized window.

  ❑   **2.2.3.3** Move a few windows by right-clicking on the title bar of the window holding the mouse down and dragging it. Do not restore window position.

  ❑   **2.2.3.4** Resize a few windows by positioning the cursor over an edge of the window, right–clicking on the mouse, holding it down and dragging it. Do not restore window size.

❑   **2.2.4** The following **Window** pull-down menu options are available: **Cascade, Tile, Arrange Icons,** Window #1 (i.e. **Settings**), Window #2, #3, etc. (i.e. **Control, Status**)

Note: Minimize the "bottom" window for the next two steps.

  ❑   **2.2.4.1 Cascade** – Arranges windows not minimized in cascading pattern within GUI.

  ❑   **2.2.4.2 Tile** – Arranges windows not minimized in tiled pattern within GUI.

Note: Minimize another window for the next step. Move the minimized windows to different locations by right-clicking on the title bars, holding the mouse down and dragging each.

❑   **2.2.4.3 Arrange Icons** – Arranges windows minimized on bottom of GUI.

Note: Open all minimized windows and choose **Cascade** for the next steps.

❑   **2.2.4.4** Window #1 (i.e. **Settings**), if applicable: _____ (write window name) – Brings window #1 to foreground or "on top" of others.

❑   **2.2.4.5** Window #2 (i.e. **Control**), if applicable: _____ (write window name) – Brings window #2 to foreground or "on top" of others.

❑   **2.2.4.6** Window #3 (i.e. **Status**), if applicable: _____ (write window name) – Brings window #3 to foreground or "on top" of others.

❑   **2.2.4.7** Window #4, if applicable: _____ (write window name) – Brings window #4 to foreground or "on top" of others.

❑   **2.2.4.8** Window #5, if applicable: _____ (write window name) – Brings window #5 to foreground or "on top" of others.

❑   **2.2.4.9** Window #6, if applicable: _____ (write window name) – Brings window #6 to foreground or "on top" of others.

❑   **2.2.4.10** Window #7, if applicable: _____ (write window name) – Brings window #7 to foreground or "on top" of others.

❑   **2.2.5** The following **Help** pull-down menu options are available: **Contents**, **Using Help**, **About…** and **Device Specifications**.

❑   **2.2.5.1 Contents** – Opens device specific on-line help file.

❑   **2.2.5.1.1 Exit** from the **File** menu or standard *Microsoft* close (**X**) button in the upper right-hand corner of the *Help* window

❑   **2.2.5.2 Using Help** – Opens general on-line help file.

❑   **2.2.5.2.1** Click on **Cancel** or standard *Microsoft* close (**X**) button in the upper right-hand corner of the *Help Topics: Windows Help* window.

❑   **2.2.5.3 About** – Displays informational dialog box containing application icon, application name, version number and last modification date.

❑   **2.2.5.3.1 What's New** – Displays informational text file in *Notepad*. Text file should describe application and its recent modifications.

❑   **2.2.5.3.1.1 Exit** from the **File** menu or standard *Microsoft* close (**X**) button in the upper right-hand corner of the *Notepad* window.

❑   **2.2.5.4 Device Hardware Specifications** – Displays informational text file in *Notepad*.

❑   **2.2.5.4.1 Exit** from the **File** menu or standard *Microsoft* close (**X**) button in the upper right-hand corner of the *Notepad* window.

❑ **2.3.0** The following tool buttons are among those available: **Print Preview, Print, Cascade, Tile,** Window #1 (i.e. **Settings**), Window #2, #3, etc. (i.e. **Control, Status**), **Device Specifications.** The last button below is an example and there will be only one for each GUI. The number represents which device is being controlled.

<div align="center">**Tool buttons**</div>

❑ **2.3.1** Each tool button has an appropriate Tool Tip displayed when cursor is positioned over button: **Print Preview, Print, Cascade, Tile,** Window #1 (i.e. **Settings**), Window #2, #3, etc. (i.e. **Control, Status**)

❑ **2.3.2 Print Preview** – Displays open file in a text viewer as currently displayed in graphical user interface (GUI).

<div align="center">**Print Preview tool button**</div>

  ❑ **2.3.2.1** Click on **Close**.

❑ **2.3.3 Print** – Prints open file by making choices in displayed a dialog.

<div align="center">**Print tool button**</div>

  ❑ **2.3.3.1** Click on **Cancel**.

Note: Minimize one window and move and resize a few of the open windows for the next steps.

❑ **2.3.4 Cascade** – Arranges windows not minimized in cascading pattern within GUI.

<div align="center">**Cascade tool button**</div>

❑ **2.3.5 Tile** – Arranges windows not minimized in tiled pattern within GUI.

<div align="center">**Tile tool button**</div>

Note: Open all minimized windows and choose **Cascade** for the next steps.

❑ **2.3.6** Window #1 (i.e. **Settings**), if applicable: _____ (write window name) – Brings window #1 to foreground or "on top" of others.

<div align="center">**Settings tool button**</div>

❑ **2.3.7** Window #2 (i.e. **Control**), if applicable: _____ (write window name) – Brings window #2 to foreground or "on top" of others.

<div align="center">**Control tool button**</div>

❑ **2.3.8** Window #3 (i.e. **Status**), if applicable: _____ (write window name) – Brings window #3 to foreground or "on top" of others.

<div align="center">**Status tool button**</div>

❑ **2.3.9** Window #4, if applicable: _____ (write window name) – Brings window #4 to foreground or "on top" of others.

❑ **2.3.10** Window #5, if applicable: _____ (write window name) – Brings window #5 to foreground or "on top" of others.

❑ **2.3.11** Window #6, if applicable: _____ (write window name) – Brings window #6 to foreground or "on top" of others.

❑ **2.3.12** Window #7, if applicable: _____ (write window name) – Brings window #7 to foreground or "on top" of others.

❑ **2.3.13 Device Specifications** – Displays informational text file in a text viewer. (The button below is an example.)

<center>**Device Specifications tool button**</center>

2.4.0 Specific controls – For specific functionality that should be provided in a particular GUI, see the requirements document for the specific device being controlled. In some cases, controls are dependent on other controls; i.e. controls may not be enabled until another control is chosen. Again, see the requirements document. The following are the *possible* controls available, and *in general*, should operate in the following manner by clicking the mouse on them:

Note: Make sure the status bar is displayed (see steps **2.2.2.2** and **2.2.2.2.1**).

❑ **2.4.1 Apply Now** (if applicable) – In the **Settings, Control** or similar window(s) this pushbutton is enabled.

<center>**Enabled Apply Now button**</center>

❑ **2.4.1.1** Click on **Apply Now** (if applicable) – A busy cursor appears on the screen and the status bar will indicate that changes to parameters in the **Settings** (or similar) window are being sent to device.

❑ **2.4.1a Load New** (if applicable) – In the **Settings, Control** or similar window(s) this pushbutton is enabled.

<center>**Enabled Load New button**</center>

❑ **2.4.2 Reset** (if applicable) – In the **Settings, Control** or similar window(s) this pushbutton is enabled.

<center>**Enabled Reset button**</center>

❑ **2.4.2.1** Click on **Reset** (if applicable) – A busy cursor appears on the screen and the status bar will indicate that a reset command is being sent to device.

❑ **2.4.3** Switch (if applicable) – In the **Settings, Control** or similar window(s) this switch is enabled (i.e. not grayed out and may be **Off**).

<center>**Enabled Switch**</center>

❑ **2.4.3.1** Click switch (if applicable) – It changes from up (**On**) to down (**Off**) or vice-versa. In some instances, a busy cursor appears on the screen and the status bar will indicate that an on/off command is being sent to device.

❑ **2.4.4** Edit box (if applicable) – In the **Settings, Control** or similar window(s) this control is enabled unless it is dependent on another selection.

<center>**Edit box**</center>

❑ **2.4.4.1** Spin (Up/down) control (if applicable) – Used with some edit boxes, this control is enabled with its associated edit box.

<center>**Spin control**</center>

❑ **2.4.4.1.1** When the up-arrow portion of the spin control is clicked the parameter in the associated edit box increments. To change the digit incrementing, highlight the digit and click the left mouse button.

❑ **2.4.4.1.2** When the down-arrow portion of the spin control is clicked the parameter in the associated edit box decrements. To change the digit decrementing, highlight the digit and click the left mouse button.

❑ **2.4.5.** Radio buttons (if applicable) – In the **Settings, Control** or similar window(s) these controls are enabled unless it is dependent on another selection.

<center>**Radio button group**</center>

❑ **2.4.5.1** When a radio button is selected/clicked, the other radio buttons in that group are de-selected.

❑ **2.4.6** Check boxes (if applicable) – In the **Settings, Control** or similar window(s) this control is enabled unless it is dependent on another selection.

<center>**Check box**</center>

❑ **2.4.7** Combo box (if applicable) – In the **Settings, Control** or similar window(s) this control is enabled unless it is dependent on another selection.

<center>**Combo box**</center>

❑ **2.4.7.1** Click on down-arrow and the combo box should open with a selectable list. The user should not be able to enter data in the box:

<center>**Open Combo box**</center>

Note: In general, the user should not be able to type data into a combo box.

❑ **2.4.8** Keyboard control – Use the **Tab** key to change control selection in the **Settings, Control** or similar window(s). Selections should change in a general top-to-bottom, left-to-right pattern.

❑ **2.4.9 Status** window(s) parameters may update/change periodically. If not now, resize the **Status** window so that all the parameters are visible. Then resize the **Settings, Control** or similar window(s) and change a parameter that is also displayed in the **Status** window. Click the **Apply Now** button and it should change in the **Status** window.

❑ **2.4.10** Status light (if applicable) – In the **Status, Control** or similar window(s) this is green or red:

<center>**Status Lights**</center>

❑ **2.5.0** The following **File** option is among those available: **Save**

❑ **2.5.1 Save** – Make a change to one of the parameters in the **Settings** (or similar) window. Choose **Save**. User will be prompted to save file to a new name in a **SaveAs** dialog box.

❑ **2.5.1.1** Click on **Cancel.**

❑ **2.6.0** The following **File** option is among those available: **Close.**

❑   **2.6.1 Close** – Closes open file and, in this mode, also terminates application. User may be asked to save file (see above).

❑   **2.7.0** Repeat step **2.1.0**.

❑   **2.8.0** The following tool button is among those available: **Save As.**

    ❑   **2.8.1 Save As** – Make a change to one of the parameters in the **Settings** (or similar) window and make a note of that change.
    ❑   **2.8.2** Click on **Save As** tool button.

<div align="center">

Save As tool button

</div>

Parameter: _____   Changed to: _____

    ❑   **2.8.2.1** Enter "**Test**" for file name in window and click **Save**.

       ❑   **2.8.1.1** If message asks to overwrite an existing file, answer **Yes**.

❑   **2.9.0** The following **File** option is among those available: **Exit.**

    ❑   **2.9.1** Choose **Exit** – Terminates application. User may be asked to save file (see above).

❑   **2.10.0** Check if parameter was actually saved. Repeat step **1.1.0** (**Local Edit**).

    ❑   **2.10.1** Repeat step **1.10.1** (**Open** a saved file.).

    ❑   **2.10.2** Check parameter changed in step **2.8.1** (**Save As**) and make sure it is what was saved.

❑   **2.11.1 Exit** application. Repeat step **2.9.1**.

Note: The GUI has now been completely checked in Local Control. This concludes the procedure.

Appendix E: Programmable Telemetry Processor (PTP) Automation

PTP Automation Installation
PTP automation requires some extra configuration of the PTP and the controlling node during installation of the automation system. The PTP must be configured so recorder module data files can be copied to the node and the PTP can be left in a near startup state when the pass is complete.

Configure the PTP for Automation

(1) To copy the data file from the PTP to the node, the assigned PTP must have the proper sharenames and the guest must be enabled.
- (a) To create the "PTP-C" sharename for drive C:
  - ✓ right click on C: in explorer
  - ✓ click sharing
  - ✓ click New Share
  - ✓ type PTP-C
  - ✓ click Permissions and set to Everyone   Full Control
  - ✓ click Add
  - ✓ click OK
  - ✓ click Apply
  - ✓ click Ok
- (b) To create the "PTP-D" sharename if the PTP has a D drive:
  - ✓ right click on D: in explorer
  - ✓ click sharing
  - ✓ click New Share
  - ✓ type PTP-D
  - ✓ click Permissions and set to Everyone   Full Control
  - ✓ click Add
  - ✓ click OK
  - ✓ click Apply
  - ✓ click Ok
- (c) To enable the guest:
  - ✓ click Start, Programs, Administrative Tools
  - ✓ click User Manager
  - ✓ double click Guest
  - ✓ uncheck Account Disabled
  - ✓ click Ok
  - ✓ Exit User Manager Window

(2) The PTP should have a desktop called NOTHING.DTP in its c:\PTP_User\Desktops directory. The node code tells the PTP to load NOTHING.DTP (if it exists in the correct directory) as the last command in the takedown section. The Nothing.DTP desktop has no modules, so by loading Nothing.DTP the PTP is left in a near startup state. When the next desktop is loaded the PTP will not have to close modules before opening the modules of the desktop being loaded. Less strain and quicker response.

Configure the Node for PTP Automation

(1) The node code needs to know the IP address of the SAFS server in order to transfer data files to SAFS. If the SAFS server is not configured in the Domain Name Server, the node must have a C:\WINNT\system32\drivers\etc\lmhosts file that contains a list of SAFS servers and their IP addresses.
Administrator privileges are required to create or change this file. Each entry should be kept on an individual line. The IP address should be placed in the first column followed by the corresponding

computername. The address and the comptername should be separated by at least one space or tab. The "#" character is generally used to denote the start of a comment.

(2) The node must have a C:\PTP_User\Logs directory to store a copy of the recorder data file produced by the PTP.

   (a) At takedown the PTP file named in the recorder module structure is copied to the node. The node copy is named using SAFS naming convention and is transferred to SAFS. If the transfer is successful the copy is deleted from the node C:\PTP_User\Logs directory, otherwise the data file stays on the node computer until it is manually removed.

   (b) For each recorder module in a stream definition three files are produced. The PTP creates and propagates the .dat file. The node software creates corresponding metadata (.mta) and flag (.flg) files on C:\Node. The metadata is a summary of the statistics for the data stream; selected Serial Input and Virtual Channel Processor information is reported if these modules appear in the data stream (See What is Metadata? for an example of a metadata file). The flag file contains the names of the data and metadata files.

   (c) If the data and metadata files are successfully transferred the flag file is also transferred.

   (d) If all files are successfully transferred, the three files (.dat, .mta, and .flg) are deleted from the node computer. If the data, metadata, and/or flag files are not transferred all three files remain on the node computer until they are manually removed.

Mission information needed to automate a PTP

Several pieces of information must be obtained from the Interface Control Document (ICD) to correctly automate the PTP and distribute the statistics to the customers. The information can be broken into three categories: information needed to create a desktop, information that describes the desktop streams, and information that describes the pass.

DESKTOP

All the information required to configure each module of a working desktop. This desktop could also be used manually.

PTP CONFIGURATION/PROFILE

This file is used at Setup to set the ping state, load the PTP desktop, and define the data streams. In order to create this file, the desktop must have been created. It is recommended that this file be created as soon as the operational desktop is complete and named the same as the desktop only with a .ptp extension. The information in this configuration file must accurately describe the data streams in the desktop. The node code will check the module type associated with a module number. If the module type defined does not match the module type registered by the PTP, the node will send an error message and setup will fail. The PTP configuration file must be created and edited with the PTP GUI since the stream definitions are binary.

(1) Ping State - whether the ping should be enabled or disabled?

(2) Desktop Name - which desktop to load onto the PTP for the pass.

(3) Stream Definitions - identify data streams, their type, and which modules define the stream.

   (a) NAME - open text describing stream content, use, owner etc. Reported in Metadata file.

   (b) TYPE - translates into a 1-character field for use by WOTIS and high level status. There should be at most one Commanding stream, at most one TM Non CCSDS stream, and at most one TM CCSDS 1 stream. If there are more than one of each of these stream types only the last stream will be used to report summary information to WOTIS. The other six stream types may be repeated within a desktop. Stream types are:

Base Band 1	(A)
Base Band 2	(B)
Commanding	(C)
Command Echo	(E)
TM Non CCSDS	(R)
TM CCSDS 1	(T)
TM CCSDS 2	(U)
Subcarrier 1	(S)

<div align="center">Subcarrier 2   (V)</div>

    (c) TAG - used to create a complete filename for SAFS. If PTP recorder module files are to sent to SAFS, SAFS must have a subdirectory matching the TAG value under the satellite ID directory. Only crucial to streams producing files to be sent to SAFS.

    (d) MODULES IN STREAM - repeated for each module in the stream.

          (i)   Module Number - use desktop as reference.

          (ii)   Module Type - use desktop as reference.

METADATA

This information is reported on the metadata file created for every recorder module contained in a stream definition. The metadata file is created even if the data files are not sent to SAFS. The Information needed:

- ✓ SPACECRAFT ID
- ✓ TR CODE
- ✓ SCHEDULED START DATE
- ✓ SCHEDULED STOP DATE
- ✓ ORBIT NUMBER
- ✓ GENERATION SITE
- ✓ PTP ID

PTP GUI

DataStreams

The DataStreams window is used to create or modify the PTP configuration/profile file. Stream definitions are binary, so modification of PTP configuration files containing stream definitions must be done using the PTP GUI. Do not use a text editor.

(1) Ping Enabled – Check box whose default is to disable the ping.

(2) Desktop – Name of the desktop to load on the PTP. Directory path C:\PTP_User\Desktops is expected.

(3) Stream Definition – Buttons are provided to help the user move between streams and create, delete, or modify the stream definitions. Definition of streams breaks down into two areas: identifying the stream and describing its contents. The engineer must provide this information. The desktop configuration defines the streams, but the stream definitions route the data to the user.

    (A) Stream Identification

        (a) Name – Identification reported on metadata file.

        (b) Type – Used to identify Commanding, TM Non CCSDS, and TM CCSDS 1 information for high level status. Also reported on the metadata and log files.

        (c) Tag - Used as subdirectory in SAFS path and part of SAFS filename.

    (B) Stream Contents – Use the desktop as a Guide.

        (a) Module Number – Desktop module number.

        (b) Module Type– Must match desktop or CheckStreamDef will produce an error and setup will fail.

Control

This window gives back the control lost by CONSOLE.EXE when the node has control of port 4000. Operator can enable or disable modules and zero counters.

Status

CONSOLE.EXE in "Receive Multicast Status" mode supplies the user with detailed module status and configuration information. This program is written by AVTEC. When the console software becomes stable and AVTEC gives its approval, a CONSOLE window will come up with the DataStream and Control windows when the PTP GUI is started. Until then, CONSOLE.EXE should be run on a PTP not being used for automation.

Automated Pass

This section describes what happens to the PTP during an automated pass and what is needed to transfer the required files to SAFS. It also specifies what is necessary for a PTP to be remotely assigned, setup for an automated pass, and returned to startup condition after a pass.

PTP assignment

(1)   Desired PTP must be turned on.
(2) A server (black window) with port 4000 in "Waiting" (See "Remote Control" line about 5 lines form the top of the window).
> The node expects to connect to port 4000, so a server with port 4001, 4002, etc. is of no use. Only one user can be in control of a PTP server port at a time, if another user is in control of port 4000 the node code can't gain control and will wait for control to be released.
> After the node has taken control "Waiting" will change to the IP address of the node computer. If port 4000 is not "Waiting" the computer with the IP address shown after "Remote Control" is the source of control for that server.
> CONSOLE.EXE control can be released by clicking "System" "Release Control" in the console window that is in control of the server. If CONSOLE.EXE is in control of a PTP server, "Connection" followed by the port and IP address of the server appears in the title bar of the window. A CONSOLE.EXE with "Status" in the title bar can only receive information from the PTP server identified by the port and IP address.
> Clicking the "x" on the PTP_NT Server title bar is an unfriendly but effective way to make sure no one has control of that server. A new PTP_NT.EXE would have to be started, but no one would be in control of the new server.
(3) Avtec PTP must be registered in the GRM reg files.
(4) If the desired PTP has port 4000 in "Waiting" on its server and is in the GRM registry file when the device GRM is started, the node code will attempt to take control of the PTP.
> If the GRM is successful the "Waiting 4000" on the PTP server will change to the IP address of the node computer. If port 4000 is not in "Waiting" when the device GRM starts, the device GRM will not reach its ready state because the ptpConnectToServer command is waiting for someone to release control of the PTP before it can execute the connect command (see step 2). The device GRM may appear to be hung, check the PTP server to see who is in control.
(5) When the device GRM is ready, the operations GRM is started.
> If the operations reaches its ready state and the PTP does not failed reset, the PTP is assigned.

PTP Setup

(1)   Make sure desired desktop is in the correct directory on the PTP.
> Expected directory is: C:\ptp_User\desktops
(2)   The desired PTP has been assigned.
(3)   Make sure the correct PTP configuration/profile file is used.
> Profile includes (minimum) the desktop name as the argument to LoadProfile. SetStreamDef commands follow the LoadProfile command. If the desktop is not on the requested PTP in the requested directory, the PTP will not be able to load the requested desktop and Verify will fail. Verify will also fail if the stream definitions module types do not match the module type information obtained by the node code when the desktop was loaded. If no streams are defined there will be no high level status information or module status reported to the log file. If stream definitions are included the PTP configuration must be created using the PTP GUI, the stream definitions are binary and a text editor will not work.
> The name of the desktop currently loaded on the PTP is included in the CheckStreamDef error message starting with ATS release 2.0. If DEFAULT.DTP or NOTHING.DTP is listed as the current desktop, the requested desktop is probably not in the proper directory on the PTP. If the current desktop matches the name of the desktop requested, one or more of the module types in the stream definitions are incorrect.
(4) The PTP configuration file is read and the embedded functions are executed.

The first function executed will probably be SetPingState (See To Ping or Not to Ping?) which enables or disables the PTP ping. LoadProfile loads the corresponding desktop on the PTP. SetStreamDef is executed for each stream defined.

PTP Start

(1) Enable all modules.
(2) Obtain system time to report actual start time on metadata file.

PTP Stop

(2) Disable all modules.
(2) Obtain system time to report actual stop time on metadata file.

PTP Takedown

Variations in the amount of time needed to complete takedown correspond to the time required to transfer the SAFS files.

(1) GetDataParameters to obtain module types and file names.
(2) Cycle through the modules, if recorder module close data file.
(3) Poll all the modules in case status has been turned off.
(4) Wait 15 seconds to make sure all modules have been polled.
(5) GetDataParameters again to obtain current module status.
(6) Set PTP sharename.
(7) If there are recorder modules and SAFS ID > 0, try to connect to local SAFS (LMHOSTS).
(8) Write log file by stream.
    (a) Write requested module information to log file.
    (b) If module type is recorder, create metadata file.
        (i) Copy the data file from the PTP.
        (ii) Transfer the data file to SAFS if connected.
        (iii) Create metadata file.
        (iv) Transfer metadata file if SAFS connected.
        (v) Create flag file.
        (vi) Transfer flag file if SAFS connected.
        (vii) Delete data, metadata and flag files from Node if they were transferred.
(6) Close SAFS connection if applicable.
(7) Load Nothing.DTP (has no modules so PTP is in near startup state).

Send SAFS files

SAFS files will be transferred if all the following conditions exist.

(1) SAFS is chosen as a resource.
(2) SAFS is up and running with proper connections
(3) SAFS directories are available.
    QuikSCAT science data QST1998212245PTP1SCI.dat would be sent to directory \QST\SCI where SCI is the "tag" of the stream definition and QST is the satellite ID.
(4) The PTP configuration file has at least one stream definition that includes a recorder module.
(5) The requested PTP has a directory path that matches the path used in the recorder module configuration.
(6) Automated PTP and node complete takedown.

What is Metadata?

Metadata files are created at TakeDown for defined data streams that contain a recorder module. The metadata files are created on the node computer and are named with the convention: PrjIDYYYYMMDDHHMMSSP##typ.mta (example: QST19980812202020P01SCI.mta corresponding to QuikScat Science data recorded on PTP 01 on 8/12/98 at 20:20:20 GMT, project/satellite ID is QST and the stream definition tag is SCI). The corresponding data file, the metadata file, and a corresponding flag

file (contains the names of the data and the metadata files) are transferred to SAFS directory (example: \QST\SCI). If the transfers are successful the three files are deleted from the node computer. If any of the transfers fail the files remain on the node computer and the failure is logged. The metadata file contains a header that describes the file followed by Serial Input and CCSDS Virtual Channel Processor statistics if that module type is in the stream definition. The stream definitions are supplied in the PTP configuration file along with the desktop used.

METADATA FILE CONTAINS:
datRemoteFilename SUMMARY
FILE SIZE BYTES: dataFileSize
FILE DATE: dataFileDate
SPACECRAFT ID: satelliteId
MODE: trCode
SCHEDULED START DATE: schStartTime
SCHEDULED STOP DATE: schStopTime
ACTUAL START DATE: actualStartTime
ACTUAL STOP DATE: actualStopTime
ORBIT NUMBER: orbitNumber
GENERATION SITE: siteId
PTP ID: PTP_unit
DESKTOP: desktopName
STREAM NAME: streamName
STREAM TYPE: streamType

MODULE NUMBER: mm Serial Input
FRAMES RECD: framesReceived
FRAMES READ: framesRead
SYNC BIT ERRORS: syncPatternBitErrors
DROPOUTS: dropouts
SLIPS: slips
LONG FRAMES: longFrames
SHORT FRAMES: shortFrames

MODULE NUMBER: mm CCSDS Virtual Channel Processor
VCID: nn
RECEIVED COUNT: vcReceivedCount[nn]
TRANSMIT COUNT: vcTransmitCount[nn]
SEQUENCE ERRORS: vcSequenceErrs[nn]
CRC ERRORS: vcCRCErrs[nn]
CORRECTABLE RS ERRS: vcCorrectableRsErrs[nn]
UNCORRECTABLE RS ERRS: vcUnCorrectableRsErrs[nn]

datRemoteFilename END OF SUMMARY

To Ping or Not to Ping?
The switch to enable or disable the PTP is a check box on the PTP GUI DataStreams window used to create the PTP configuration file. The default is to disable the ping. If the box is checked the ptpEnablePingEx command will be sent to the PTP at setup when the configuration file commands are executed; otherwise the ptpDisablePingEx command will be executed.

ORIGINAL PROBLEM
If sockets are opened in block mode and the transmit side is open, a broken network connection is detectable. However if in read only block mode (which all PTPs are) and the network connection is broken, it will be approximately six days before the server knows the client went away.

AVTEC'S SOLUTION TO THE ORIGINAL PROBLEM

When a TCP connection on a recorder socket is opened the PTP detects logins and watches for traffic. If no traffic in 20 seconds, PTP sends a ping to the standard ping port. If no response within 20 seconds, reinitialize the connection. This is the ping.

PROBLEM CREATED BY THE FIREWALL

The firewall prevents the ping from reaching its destination so the software never receives a response and so a reconnect is initiated every 40 seconds.

PTP SOLUTION TO THE FIREWALL PROBLEM

Avtec has created a PTP function to disable the ping created to test the network connection. This prevents the command socket time outs and reconnects, but the PTP is not checking for network disconnect.

CONTROL STATUS IS BEING HANDLED

The AWOTS software performs a status check at a given rate. Within the status check for the PTP is a GetNoOp command that checks if the node code is still connected to the PTP. This check lets the node code know it is still in control but doesn't give any information about the status of the data or command connections.

DISABLE OR ENABLE PING

If the firewall will not block the pings, it is better to have the pings enabled. However if the firewall will block the data and/or command pings, these sockets will be disconnected and reconnected approximately every 40 seconds unless the ping is disabled. Avtec has written three functions to handled the ping: ptpDisablePingEx, ptpEnablePingEx, and ptpGetPingStateEx. These three functions have been incorporated into the PTP code and the PTP configuration file.

How to Use the Desktop to Create a PTP Configuration/Profile File.

The stream definitions should be defined in the Interface Control Document (ICD), if they are not the PTP desktop can be used to create a crud configuration file. Use CONSOLE.EXE to load the desktop. Make a note of the modules in the desktop and then use "Edit Module Connections" under "Remote Desktop" to determine how the modules are connected. The desktop will only reveal which modules were used and how they were connected. An operator may be able to determine the stream type, but only the Project Managers know how they want the streams labeled.

The Node Can't Copy the PTP Data File.

If the PTP drive is shared correctly and the recorder module path is correct but the copy command is returning "Win32 error: 1326 Logon failure: unknown user name or bad password.", check if the node and the PTP are in the same Domain. Increased security may not let allow the node to copy a file from the PTP if their logon domains are different.

Appendix F: Station Status Broadcasts

1. Introduction

The Automated Tracking Software (ATS) is a multi-process Windows NT software system that contains several core applications. One of these applications is *MonitorAndControl.exe (M&C*) located under the folder *c:\Master* on any EPGN Master PC. This application is responsible for receiving equipment status collected on a Master and sending it to remote clients using a TCP/IP socket connection on the Control and Status sub-network maintained by Goddard Space Flight Center (GSFC). The status is sent to remote clients using the *StationStatusMonitor* thread executing under *M&C*. Successful startup of this thread at support initialization times is required to fulfill *Reporting Master* responsibilities.

These status messages can be displayed using the ATS utility *StationStatusDisplay.exe* (SSD) also located under the folder *c:\Master*. Status can also be processed on the remote client end using any other software application that can connect to the master executing SSB and read and decode the series of messages.

All station status messages are formatted with respect to the byte order used on Intel PC platforms. No byte-swapping to accommodate processing and display on remote clients with other operating systems is done.

2. Operational Design

The *MonitorAndControl.exe* core process on the EPGN Master collects and displays status from all ATS-configurable devices. An M&C threaded function, *StationStatusMonitor (SSM)*, relays the status to remote clients identified in the support profile using TCP/IP socket connections. The following sequence of operator and ATS actions will initiate the socket connections and start the transmission of status messages:

(1) Remote client operators start their software and begin "listening" for a remote TCP/IP socket from the *StationStatusMonitor* thread executing in *MonitorAndControl*. The remote client software must be executing prior to support initialization.

(2) At support initialization time (typically AOS – 10 minutes), the *StationStatusMonitor* thread reads IP addresses and port numbers from an operator-defined list contained in the support or operational profile.

(3) *SSM* begins sending a *heartbeat* message from an EPGN Master to remote clients. The format of the heartbeat message, as well as other messages, is described below in **3. Status Broadcast Message Format**. This *heartbeat* message verifies the communication integrity between the Master and remote client.

(4) At support *Initialization*, typically ten minutes prior to AOS, *SSM* is started in M&C. *SSM* formats the status and relays it to the remote clients identified in the profile. This status relays occurs between the *Initialization* and *Takedown* phases of the ATS automation cycle. The *StationStatusMonitor* thread terminates and this and the TCP/IP sockets are closed at *Takedown..* See **Figure 1** for the general design of the *StationStatusBroadcaster.exe* application.

3. Status Broadcast Message Format

The *StationStatusBroadcast* application fills a data structure with three 4-byte integers and a character string that will never exceed 600 bytes. The following C data structure describes the format of the status message:

```
struct StatusData {
    int iSite;   // site ID
    int iType; // message type
```

```
     int iSize;  // message size
    char sStatus[600]; // message body
}; // 612-byte structure
```

StatusData statusdata; // declare statusdata of type StatusData

The **site ID** is sent as a 4-byte integer. The following table defines the values that may be sent for the site ID and their corresponding sites.

ID	site
0	AGS
1	MGS
2	SGS
3	WGS

Note: Execution of the SSB process on each reporting EPGN Master can be achieved. Remote client display applications should be developed to accommodate this possibility.

The **message type** is also transmitted as a 4-byte integer value. The following table defines the values that may be sent for the message type and their corresponding descriptions. The **message type** value and the contents of the **message body** are directly related. The contents of each **message type** are discussed in **4. Status Broadcast Message Contents**.

Type	Description
0	Heartbeat
1	Initialization
2	Status
3	Schedule
4	Takedown

The **message size** is an integer value that indicates the size of the status message being transmitted. This value is the byte sum of the three 4-byte integers and the variable length character string status **message body**. The following line of C source code fills the data structure **message size** *iSize* data member:

statusdata.iSize = strlen(statusdata.sStatus) + 12;

The **message body** is a variable length character string that will never exceed 600 bytes. Contents of the **message body** are discussed in **4. Status Broadcast Message Contents**.

4. Status Broadcast Message Contents

Heartbeat (message type = 0) status messages are transmitted upon successful startup of *StationStatusBroadcaster.exe* on the Master PC. Successful startup of this process will begin filling the status data structure with the three 4-byte fixed words and time as reported on the Master since equipment status monitoring is active only between the *Initialization* and *Takedown* phases of the automation cycle. The heartbeat status messages are transmitted continuously until the operator terminates SSB. The following section of C source code prepares the data structure:

long lTime; // time (seconds since 1/1/1970)

```
// nameStation defined in StationAssets…
        if(nameStation[0] == 'A') stationstatus.iSite = 0;
        if(nameStation[0] == 'M') stationstatus.iSite = 1;
```

```
        if(nameStation[0] == 'S')  stationstatus.iSite = 2;
        if(nameStation[0] == 'W') stationstatus.iSite = 3;

        heartbeatstatus.iType = 0; // heartbeat

        while (TRUE)
        {
          time(&lTimeNow);

            sprintf(stationstatus.sStatus,"%ld",lTimeNow);
            heartbeatstatus.iSize = strlen(stationstatus.sStatus) + 12;

            for (iLoopSockets = 0; iLoopSockets < iNumClientSockets; iLoopSockets++)
            { // send to number of user-defined remote clients ….
              send(sockClients[iLoopSockets], (const char*)&stationstatus, stationstatus.iSize, 0);
            } // End for.
        } // End while.
```

Since a typical reported time value is nine characters long, "922831296" for March 30, 1999 at 22:03:00 GMT for instance, the heartbeat message should be 21 bytes long (12 fixed bytes + 9 status).

The **Initialization** message consists of the standard 12 fixed bytes, as well as the variable length ATS satellite operational profile name loaded during the support. Only one **Initialization** message is sent per support. It marks the beginning of the automation cycle when profile-requested devices are requested for assignment. **Initialization** of the QuikSCAT profile, "QST_ 03_30_1999_21_26_29_00" for instance, will force SSB to transmit 12 + 27 = 39 bytes in order to accommodate the profile name in the status word.

Note: Operational support profiles are named using the following convention,
SatNam_MO_DA_YYYY_HH_MM_SS_AN.

where,
   SatNam = WOTIS satellite Identifier (QuikSCAT designated as "QST")
   MO = 2-digit month
   DA = 2-digit day-of-month
  YYYY = 4-digit year
   HH = 2-digit hour (GMT)
   MM = 2-digit minute
   SS = 2-digit second
   AN = 2-digit receive antenna (11, for the 11-meter antenna)

The **status** message consists of a character string that includes a representative label and value for the equipment type and unit monitored for station broadcasting purposes. **A status message will contain information for only one instrument. The concatenation of all EPGN instrument status into one character string is no longer supported in SSB**. If no status is available for a monitored unit, no status for that unit is included in the string. The following table defines the labels and values that may be included in the string. In cases where more than one unit of a type of equipment is monitored, the unit number is appended to the label.

Label	Status Description	Format
"XAGC"	SCC X-Band Signal Strengths	"aa.a (dB) bb.b (dB) cc.c (dB) dd.d (dB)"
		(where a, b, c and d are channels 1-4, respectively)
"XRcvAGC"	SCC X-Band Tracking Receiver Sig.Strength	"ss (dB)"
"XDemod1"	SCC X-Band (SA924) Demodulator Status (unit 1)	"0" (UNLOCK) or "1" (LOCK)

"XDemod2"	SCC X-Band (SA924) Demodulator Status (unit 2)	"0" (UNLOCK) or "1" (LOCK)
"XDemod3"	SCC X-Band (SA924) Demodulator Status (unit 3)	"0" (UNLOCK) or "1" (LOCK)
"XDemod4"	SCC X-Band (SA924) Demodulator Status (unit 4)	"0" (UNLOCK) or "1" (LOCK)
"XBitSync1"	SCC X-Band (SA924) Bit Sync Status (unit 1)	"0" (UNLOCK) or "1" (LOCK)
"XBitSync2"	SCC X-Band (SA924) Bit Sync Status (unit 2)	"0" (UNLOCK) or "1" (LOCK)
"XBitSync3"	SCC X-Band (SA924) Bit Sync Status (unit 3)	"0" (UNLOCK) or "1" (LOCK)
"XBitSync4"	SCC X-Band (SA924) Bit Sync Status (unit 4)	"0" (UNLOCK) or "1" (LOCK)
"SRcvAGC1"	SCC S-Band Receiver LHC Data Signal Strength	"ss (dB)"
"SRcvAGC2"	SCC S-Band Receiver RHC Data Signal Strength	"ss (dB)"
"SRcvAGC3"	SCC S-Band Receiver LHC Tracking Signal Strength	"ss (dB)"
"SRcvAGC4"	SCC S-Band Receiver RHC Tracking Signal Strength	"ss (dB)"
"SBitSync1"	ATS Decom7715 Bit Sync (unit 1) status	"0" (UNLOCK) or "1" (LOCK)
"SBitSync2"	ATS Decom7715 Bit Sync (unit 2) status	"0" (UNLOCK) or "1" (LOCK)
"SBitSync3"	ATS Decom7715 Bit Sync (unit 3) status	"0" (UNLOCK) or "1" (LOCK)
"SBitSync4"	ATS Decom7715 Bit Sync (unit 4) status	"0" (UNLOCK) or "1" (LOCK)
"SBitSync5"	ATS Decom7715 Bit Sync (unit 5) status	"0" (UNLOCK) or "1" (LOCK)
"SBitSync6"	ATS Decom7715 Bit Sync (unit 6) status	"0" (UNLOCK) or "1" (LOCK)
"SBitSync7"	ATS Decom7715 Bit Sync (unit 7) status	"0" (UNLOCK) or "1" (LOCK)
"SBitSync8"	ATS Decom7715 Bit Sync (unit 8) status	"0" (UNLOCK) or "1" (LOCK)
"SBitSync9'	ATS Decom7715 Bit Sync (unit 9) status	"0" (UNLOCK) or "1" (LOCK)
"SBitSync10"	ATS Decom7715 Bit Sync (unit 10) status	"0" (UNLOCK) or "1" (LOCK)

Label	Status Description	Format
"SFrmSync1"	ATS GDP225 Frame Sync (unit 1) status	"0" (UNLOCK) or "1" (LOCK)
"SFrmSync2"	ATS GDP225 Frame Sync (unit 2) status	"0" (UNLOCK) or "1" (LOCK)
"SFrmSync3"	ATS GDP225 Frame Sync (unit 3) status	"0" (UNLOCK) or "1" (LOCK)
"SFrmSync4"	ATS GDP225 Frame Sync (unit 4) status	"0" (UNLOCK) or "1" (LOCK)
"SFrmSync5"	ATS GDP225 Frame Sync (unit 5) status	"0" (UNLOCK) or "1" (LOCK)
"SFrmSync6"	ATS GDP225 Frame Sync (unit 6) status	"0" (UNLOCK) or "1" (LOCK)
"SFrmSync7"	ATS GDP225 Frame Sync (unit 7) status	"0" (UNLOCK) or "1" (LOCK)
"SFrmSync8"	ATS GDP225 Frame Sync (unit 8) status	"0" (UNLOCK) or "1" (LOCK)
"SFrmSync9"	ATS GDP225 Frame Sync (unit 9) status	"0" (UNLOCK) or "1" (LOCK)
"SFrmSync10"	ATS GDP225 Frame Sync (unit 10) status	"0" (UNLOCK) or "1" (LOCK)
"BVLDS1"	ATS Metrum Recorder (unit 1) status	"SomeState-Address=Block#"
"BVLDS2'	ATS Metrum Recorder (unit 2) status	"SomeState-Address=Block#"
"BVLDS3"	ATS Metrum Recorder (unit 3) status	"SomeState-Address=Block#"
"BVLDS4"	ATS Metrum Recorder (unit 4) status	"SomeState-Address=Block#"
"BVLDS5"	ATS Metrum Recorder (unit 5) status	"SomeState-Address=Block#"
"BVLDS6"	ATS Metrum Recorder (unit 6) status	"SomeState-Address=Block#"

(for instance, "Recording-Address=12345")

"PTP1"	ATS Programmable Telemetry Processor (unit 1) status	"stream 1, stream 2, … stream n"
"PTP2"	ATS Programmable Telemetry Processor (unit 2) status	"stream 1, stream 2, … stream n"
"PTP3"	ATS Programmable Telemetry Processor (unit 3) status	"stream 1, stream 2, … stream n"
"PTP4"	ATS Programmable Telemetry Processor (unit 4) status	"stream 1, stream 2, … stream n"
"PTP5"	ATS Programmable Telemetry Processor (unit 5) status	"stream 1, stream 2, … stream n"
"PTP6"	ATS Programmable Telemetry Processor (unit 6) status	"stream 1, stream 2, … stream n"

(where n = number of streams loaded in PTP desktop)

| "AZ-Mode=" | SCC antenna control unit azimuth angle processor mode | "OFF" or "ON" or "ENABLE" |
| "EL-Mode=" | SCC antenna control unit elevation angle processor mode | "OFF" or "ON" or "ENABLE" |

| "AZ=" | SCC antenna control unit azimuth angle (degrees) | "123.456" |
| "EL=" | SCC antenna control unit elevation angle (degrees) | "123.456" |

| "ExcMode" | SCC exciter mode | "NOT_ACTIVE" or "SWEEPING" or "LOCK" |
| "ExcRcvCoh" | SCC exciter coherency | "modulation=32dB" or "modulation<32dB") |

| "EOS-SS" | SCC EOS channel signal strengths | "aa.a bb.b" |

(where a and b are channels 1 and 2, respectively)

"REC#1"	SCC X-Band recorder (unit 1)	"state frameID tapeID loaded? Tape% GMT"
"REC#2"	SCC X-Band recorder (unit 2)	"state frameID tapeID loaded? Tape% GMT"
"REC#3"	SCC X-Band recorder (unit 3)	"state frameID tapeID loaded? Tape% GMT"
"REC#4"	SCC X-Band recorder (unit 4)	"state frameID tapeID loaded? Tape% GMT"
"REC#5"	SCC X-Band recorder (unit 5)	"state frameID tapeID loaded? Tape% GMT"
"REC#6"	SCC X-Band recorder (unit 6)	"state frameID tapeID loaded? Tape% GMT"

where,

state = "PLAY, "REVERSE_PLAY", "RECORD", "STOP",
"FAST_FORWARD", "REWIND", "SEARCH", "UNKNOWN", or
"OFF_LINE"
frameID = tape frame ID number
tapeID = 10-character tape name
loaded? = tape loaded state ("UNLOADED" or "LOADED")
Tape% = "TapeUse=%"; where % = percentage (of tape used * 100)
GMT = "day-of-year hour:minute:second"; "089 23:59:59.123", for instance.

Label	Status Description	Format
"SbandDataCombiner"	SCC S-Band Data Combiner	"UNLOCK or LOCK" for
		Carrier Detect, PM Demod, and BPSK Demod
"SbandTrackCombiner"	SCC S-Band Track Combiner	"UNLOCK or LOCK" for
		Carrier Detect, PM Demod, and BPSK Demod

A **Schedule** type messages is transmitted at each phase of the automation cycle in order to update the current status of the support. **Schedule** messages contain the details of the operational schedule window displayed on the EPGN Master PC. Typical **Schedule** status messages include :

"Track LS7 02150 (115) 04/29/1999 14:15:04 TR1 SUP 11-11 023 (115) 04/29/1999 14:25:00 04/29/1999 14:35:00   Waiting"

and,

"Record LS7 02150 AMPEX #1   X1   REC   (115) 04/29/1999 14:26:30 04/29/1999 14:30:45"

where,

"Track" and "Record" = scheduled task type;

"LS7 02150" = WOTIS scheduled satellite ID and orbit number;

"(115) 04/29/1999 14:15:04" = Support *Initialization* time including (day-of-year)
 "TR1 SUP 11-11 023" = TR code, schedule mode, receive-transmit antennas and SCC configuration number;

"(115) 04/29/1999 14:25:00 04/29/1999 14:35:00" = Support AOS and LOS, respectively; including (day-of-year);

"Waiting" = support state; other states include "Initialized", "Setup", "Start", "Stop", "Takedown" and "Complete";

"AMPEX #1 X1 REC" = X-band recorder type and number, scheduled X-band event and REC mode, respectively;

"(115) 04/29/1999 14:26:30   04/29/1999 14:30:45" = X-band recorder start and stop times, respectively, including (day-of-year).

The **Takedown** message marks the end of the automation cycle and device status polling. The following 33-byte character string message is transmitted at **Takedown**:
                    "Support status monitor completed."

 A 180 second sleep is introduced into the *StationStatusMonitor* thread after transmission of the **Takedown** message by SSB in order to respect ATS time delays to reset all equipment, compile post-pass summary reports and prepare for the next upcoming support.   At sleep end, another **Schedule** message type is transmitted to the remote clients in order to update upcoming events.

TCP/IP   Socket Message

Remote Client

Symbol Key

Status Message   ATS Master Process   ATS Master Thread

Figure F-1: Station Status Broadcaster General Design

Appendix G: Tape Shipping Report Format

```
Master Tape                  Conveyance or
  Source        Media ID        Box ID       Ship Date

 Field    1    5              15            25            33
          SIDRRTTTTTTTTTTBBBBBBBBBBYYYYMMDD
```

SID = 3-character ground station ID (WPS, SGS, AGS, MGS); *automatically input*.
RR = 2-character recorder type (SS for Sony, AA for Ampex); *automatically input*.

TTTTTTTTTT = 10-character master tape media identifier; *multiple list box selection*.

BBBBBBBBBB = 10-character box identifier; *operator input required*.

YYYYMMDD = year, month and day shipping file was created; *automatically input*.

Appendix H Configuring the GRM Registry Files

Introduction and overview

There are at minimum three General Resource Manager registry files that need to be configured. Each of the three types has its own section and set of instructions.

The first registry file is located on a "Master Control" computer; the second two (or more) registry files are located on (various) Node computer(s).

It is unlikely that a "Master Control" computer will have more than one registry file. The "Master Control" computer GRM is simply denoted the "Master" GRM.

Master-Node communication is executing between the GRMs. Successful communication depends on procedures, as well as the existence of the "Developer" or "Operator" accounts on the Master PC.

On the other hand, a "Node" networked computer requires two General Resource Managers and two corresponding General Resource Manager registry files. One General Resource Manager is responsible for controlling the various attached instruments. This General Resource Manager is denoted the "Device" GRM. The other General Resource Manager is responsible for tracking station "operations" and is denoted the "Operations" GRM.

The "Operations" and "Device" General Resources Managers will be chained together in a master/slave relationship with the "Operations" GRM serving the role of master for the two. In turn, this pair will be chained to yet another GRM, the "Master" GRM, which will then assume the role of "master." This "Master" GRM is most likely resident on another networked computer. The net result of all this will be at minimum a three-GRM chain distributed over two networked computers. Through judicious registry file setup the chaining of GRMs could potentially continue all across a world-wide network.

The technique of configuring a GRM registry file is to take a ready-made (template) registry file that contains just about every possible entry that would want to be known by a GRM in an Automated Tracking Station and tailor it to the specific installation of interest.

This tailoring is done in one of two ways: 1) by uncommenting various lines of text within the registry file you are editing, effectively "turning on" that portion of the registry, or, 2) by copying the appropriate lines from the template registry file to your being edited registry file and then "tuning" them to your requirements.

First, Some Vital Fundamentals

Let's introduce some crucial concept areas such as "Port", "Resource" and "Connection." A clear understanding of these as they are understood by a General Resource Manager is supremely important.

Ports For The Non-seafaring

A "port" is what a General Resource Manager uses to detect the presence of and communicate with "targets", be they a specific instrument directly connected to a networked computer or another General Resource Manager, also on a networked computer. Careful specification of "port" settings in a GRM registry file provides the General Resource Manager the "message transport machinery" to use when it starts and looks for what it anticipates will be able to be seen by itself.

For a specific instrument a "Port" entry in the "Device" GRM registry file will most often have the settings of an RS-232 style port on the computer to which the instrument is connected directly.

For the "operations" control of a type of instrument a "Port" entry in the "Operations" GRM registry file will have the name of a "pipe" that will be used to funnel messages from the "Operations" GRM to the "Device" GRM that is aware of and handling directly connected instruments.

For the "master control" of a collection of "Operations" GRMs a "Port" entry in the "Master" GRM registry file will have the name of a "pipe" that will be used to funnel messages from the "Master" GRM to a specific "Operations" GRM. Note that this is similar in appearance to that of the "operations" control style just mentioned.

Resources For The Resourceful

The concept of "Resources" is a broad and slippery topic in the design of our General Resource Manager. In its pure sense, a "Resource" can be most anything: a specific instrument listed with a designated unit number, the "operational scripting" of a collection of instruments of the same type, or yet-another-General Resource Manager. Careful specifying of the "Resources" for a particular GRM is vital when chaining one GRM to another GRM.

For a specific instrument a "Resource" entry in the "Device" GRM registry file will most often have the name of the type of instrument and a designated unit number. Note that in this model, complete computer systems can be thought of as being an instrument of a specific type.

For the "operations" control of a type of instrument a "Resource" entry in the "Operations" GRM registry file will have the name of the "Operations Scripting Class" of a specific type of instrument along with either the DNS-style name or IP address of the computer on which the "Device" GRM is located. Most frequently the "Operations" GRM and the "Device" GRM will be located on the same "Node" networked computer, but this is not an absolute requirement. There are at minimum two "Resource" entries in a registry file for an "Operations" GRM: at least one to designate which "Device" GRM(s) are to receive the flow of messages to and from this "Operations" GRM and at least one entry for an "Operations Scripting Class" of a type of instrument.

For the "master control" of a collection of "Operations" GRMs a "Resource" entry in the "Master" GRM registry file will have the DNS-style name or IP address of the networked computer on which the "Operations" GRM is located. Most often this is a separate networked computer, in contrast to that of "Operations" GRMs and "Device" GRMS being on the same networked computer, but this is not an absolute requirement. There are at minimum two "Resource" entries in a registry file for a "Master" GRM: one to designate which "Operations" GRM(s) are to receive the flow of messages to and from this "Master" GRM and at least one entry for a "Remote Node" networked computer.

The concept of a "Remote Node" computer is essentially this: to start GRM processing you need a "seed" to feed the sequence. A General Resource Manager exists in its own right. It is aware of who feeds it, along with who should be fed by it only through the interpretation of entries in its corresponding registry file.

Recall that General Resource Managers can be chained, much like a linked-list. The entry for "Remote Node" is in essence the head of a linked list of chained together sub General Resource Managers. The "Master" GRM can be considered at the top of this food chain under the control of a process that will feed this GRM.

The design of the "Master Control" software originally did not call for a "Resource Manager". The design of a General Resource Manager called for it to be totally non-dependent upon which process fed it and which processes it, in turn, fed. There needed to be a way to have a General Resource Manager become aware of the process that fed it along with the "Master Control" processes become aware of a "Resource Manger." This is handled by the introduction of the entry into the "Master" GRM registry file called "Remote Node" which provides a "source" to the GRM from which to receive messages. Observe that the "Remote Node" entry in the "Resources" section has a bracketed section with a "1" and the "." character.

This indicates that the "master control" process" that feeds the "Master" GRM resides on the same computer as does the "Master" GRM" and that the process is designated with a "ProgramId" is 1.

Connections Bring It All Together

Finally we move into the "Connections" section of a GRM registry file. As was the case with the "Ports" and "Resources" sections the "Connections" section has its own collection of things that must be known about. You will note that after scanning the "Ports" and "Resources" sections in a registry file that there is not yet any binding between an entry in the "Resource" section and a corresponding entry in the "Ports" section. It is the "Connections" section which binds these other two sections together.

For a specific instrument a "Connections" entry in the "Device" GRM registry file has two parts. The first part is the type of port and its number to which the instrument is connected. The second part is the name of the type of instrument and its designated unit number. Note that these items are pulled from the "Ports" and "Resources" section of the same registry file.

For the "operational" control of a type of instrument a "Connections" entry in the "Operations" GRM registry file also has two parts. This time the first part is the name and number of a "pipe" from the "Pipe" section. The second part is the "Operations Scripting Class" of a specific type of instrument along with the number of ***needs more explanation *** which is to flow control through the chosen pipe. There are at minimum two "Connections" entries in a registry file for an "Operations" GRM: at least one to designate which "Device" GRM(s) are to receive the flow of messages from this "Operations" GRM and at least one entry for an "Operations Scripting Class" of a type of instrument. Observe that an entry for the "Device" GRM entry will have a digit indicating the "ProgramId" of the "Device" GRM to which this "Operations" GRM is to be chained.

For the "master control" of a collection of "Operations" GRMs a "Connections" entry in the "Master" GRM registry file also has two parts. This time, as was the case with the "Operations" GRM registry file "Connections" section, the first part is the name and number of a "pipe" from the "Pipe" section. The second part is the "Operations" GRM "ProgramId" which is to receive the flow of messages from this "Master" GRM. There are at minimum at least two "Connections" entries in a registry file for a "Master" GRM: one to designate which "Operations" GRM(s) are to be chained and receive the flow of messages from this "Master" GRM and at least one entry for a "Remote Node" networked computer.

The concept of a "Connection" entry for a "Remote Node" computer is essentially this: to feed a GRM you first need a "pipe" from which the GRM is to receive its messages. Next you need to designate a "Resource" to and from which the "Master" GRM will communicate. The "Remote Node" "Connection" entry binds these two items together.

Tuning and Tweaking Your Registry Files Overview

The tune and tweak process essentially follows this sequence: get two workspaces loaded with the desired file(s), then tune and then tweak the various sections in the file you want to save.

We will configure the Master Control computer registry file first and then move on into a Node networked computer and configure the registry files for "Operations" and "Devices."

The tune and tweak starts first with "Ports". Your tune and tweak will continue with "Resources" and then continue on into "Connections." You will finish up a registry file by specifying a couple of values in the "Configuration" section.

To Configure the Master Registry File

❒ Start the Notepad text editor.
❒ Open the file "GRMBase.reg" from the c:\Master\Regs *** check this location *** directory in Notepad. You will use this file as a template to supply lines of text that will be copied to another text file.
❒ Start a second copy of the Notepad text editor. (Unlike fancy word-processors, Notepad is not able to have two files opened at the same time for editing. To open another file you must start a second copy of Notepad.)
❒ Check to see if you have a file named "GRM.reg" in your c:\Master\Regs directory. If so, then open this file into your second copy of Notepad. If not, then open the same "GRMBase.reg" file into your second copy of Notepad and then "Save As" this workspace as "GRM.reg" into the c:\Master\Regs directory.
  ❒ Note what you did to obtain your workspace copy of the "GRM.reg" file. If you opened an existing file you will do the following steps by copying lines of text from the "GRMBase.reg" into your "GRM.reg" file in appropriate areas.
  ❒ On the other hand, if you created your "GRM.reg" file by making a copy of the existing "GRMBase.reg" file, you will then proceed by uncommenting lines of text from your new "GRM.reg" file and then make the appropriate adjustments.
❒ In the "Port" section of your GRM.reg file create a pipe port for each "Node" networked computer in the tracking station by copying a line with the pipe port information from the "Port" section of the opened GRMBase.reg file. Also create one pipe port for the to-be-created "RemoteNode" resource. Number the pipe ports sequentially starting with zero. An explanation of what's going on with this "RemoteNode" is in a later section.
  ❒ Example lines for the "Port" section of the "Master" GRM registry file follow:
    ❒ \Pipe\0
    ❒ \Pipe\1
❒ In the "Resources" section of GRM.reg create a "GRM" resource for each "Node" networked computer of which you want the "Master" GRM to be aware. Make the resource entries by copying the information from "Resources" section of the GRMBase.reg file. Review the introduction above regarding "Resources" and the "Master" General Resource Manager.
  ❒ Note from the following example line that the "GRM" line has the "ProgramId" of the target General Resource Manager to and from which messages are sent and received. The bracketed section on this line contains a "unit number" and the DNS style name of the computer where the target, chained, GRM resides. You may optionally specify the target computer using the target computer's IP address.
  ❒ An example line for the "Resources" section follows:
    ❒ \GRM\2 [0 SGSNode]
❒ In the "Resources" section of the GRM.reg file create an entry for a "RemoteNode" resource by copying the resource information from the "Resources" section of the GRMBase.reg file. Observe that the "RemoteNode" example line has a unit number of "1" and a "." character indicating that the "target" machine for the "RemoteNode" resource is the same machine as where the "Master Control" GRM resides. Recall that the "." is short-hand notation for "same computer" when directly specifying a numeric IP address.
  ❒ An example line for the "Resources" section of the "Master" GRM registry file follows:
    ❒ \RemoteNode\0 [1 .]
❒ In the "Connections" section of the GRM.reg file create a "connection" entry for each "resource" by copying the connection information from the "Connections" section of the GRMBase.reg file. Review the introduction above regarding "Connections" and the "Master" General Resource Manager.
  ❒ Note from the following example lines that the first entry binds the first pipe port from the "Ports" section to the "target" GRM from the "Resources" section, whereas the second entry binds the second pipe port from the "Ports" section to the "RemoteNode" from the "Resources" section.

❑ Example lines for the "Connections" section of the "Master" GRM registry file follow:
    ❑ \Pipe\0\GRM\2
    ❑ \Pipe\1\RemoteNode\0
❑ You're almost done with the "Master" GRM registry file. To finish up the GRM.reg file you now need to go to the "Configuration" section and set the "ProgramId" to "1".
❑ Finish off "Master" tune and tweak by ensuring that the "MaxResources" entry count is sufficient for the number of resources in this GRM registry and the GRM registries on all of the "Node" network computers. Observe that you will need to visit each of the Node computer registry files first to obtain a count prior to finishing off this step with a value.

To Configure the (Node) Device Registry File

❑ The steps in this sequence are similar to the Master registry file configuration.
❑ Just as a reality check, this sequence assumes that you have moved from the "Master" computer over to another console and keyboard attached to a "Node" computer. It is possible, though highly unlikely, that both the "Master" and "Node" systems are located on the SAME computer. As of May, 1999 there is only one known instance of this and it is located at McMurdo Ground Station in Antarctica.
❑ As you did for the "Master" GRM registry file, open the template file "GRMBase.reg" from the c:\Node\Regs directory in Notepad. *** Note the different location! *** *** Is this the same file? ***
❑ Do a "Save As" and save the template file into c:\Node\Regs directory using a name of GRM#*.reg where the "#*" is the number of the combination of the "Device" GRM position in a list of chained GRMs and the number of the Node networked computer where this file resides.
    ❑ The number of your Node may be determined by establishing the position of this Node within the collection of Node networked computers within a tracking station. You will then determine where this registry file lives along a sequence of chained together General Resource Managers. If there is only one Node in a tracking station, then the position of this Node is obviously 1. If there are more Node networked computers, then they will be numbered sequentially starting with 1. The number of a "Device" GRM in a multi-node tracking station is "*#", where the "*" is the number of the Node in the list of Node networked computers in a tracking station and the "#" is the position of the "Device" GRM within the collection of chained together GRMs. If there is only one Node in a tracking station then the first digit of "1" is omitted from the GRM number and is implied.
        ❑ Example: The number of a "Device" GRM in a single Node tracking station is "3", since it is the 3rd member of the GRM chain. The 2nd member of a GRM chain is usually the "Operations" GRM which will be covered in the next section.
        ❑ Example: The "Device" GRM registry file on the 3rd Node networked computer in a tracking station would be named "GRM33.reg."
❑ Start a second copy of Notepad as you did with the "Master" GRM.reg registry file.
❑ Open the "GRMBase.reg" template registry file once again, from the c:\Node\Regs directory into the second copy of Notepad.
❑ Make a list of all of the instruments directly connected to this Node networked computer. Then establish the port setting for each instrument (baud rate, parity, stop bits, etc as appropriate) along with the COM port number. Review the table in this section that details how to specify COM port settings.
    ❑ If your instrument is connected to a Digibox multi-serial port device you will select a COM port number based upon which jack the instrument's cable is connected. Jack #1 on a Digibox corresponds to COM 3 on the Node. Jack #2 on a Digibox corresponds to COM 4 on the Node, and so on all the way up to COM 18 for a 16 jack Digibox. There may be additional chained Digibox units. The numbering sequence continues the same on up through the additional Digiboxes.
    ❑ Note which instruments are connected through any port adapters such as an IO-Tech IEEE488 to RS-232 converter. Converter-adapter devices do not have typical COM port information such as baud rate and parity.
❑ In the "Port" section of your newly created "GRM#*.reg" registry file create a "port" entry for each instrument directly connected to this Node by copying the corresponding port information for that instrument from the GRMBase.reg file. Note that the GRMBase.reg template file instrument "port" entries should have the COM port settings (baud, parity, stop bits and the like) already specified for a

selected instrument. All you should have to do for a standard RS-232 style instrument is to supply a correct COM port number.

```
Table of RS232 Settings in a "Device" Register File
```

```
Following the name and unit number of the RS232 port is a list enclosed in
brackets that specifies the serial port's baud, stop bits, data bits, parity, and
overlapped I/O. The list is in the form

[bb ss dd pp o]

where bb specifies the baud rate setting,
      ss specifies the stop bits setting,
      dd specifies the data bits setting,
      pp specifies the parity setting, and
      0  specifies the overlapped I/O flag.

The list is column dependent.

Baud Rate          Stop Bits          Data Bits          Parity
----------------   ----------------   ----------------   ----------------
Setting  Meaning   Setting  Meaning   Setting  Meaning   Setting  Meaning
  1        300       1        1         1        7         1        None
  2        600       2        1.5       2        8         2        Even
  3        1200      3        2                            3        Odd
  4        2400                                            4        Mark
  5        4800
  6        9600
  7        19200
  8        38400

Example: To specify a setting of 9600 baud, 1 Stop bit, 8 Data bits, and no
parity, use [ 6  1  2  1].
```

❒ Observe in the example lines that the COM port number follows the field for specifying what type of port is in use. The bracketed section contains the baud rate, stop bits etc for that port.

❒ Example lines for the "Port" section of the "Device" GRM registry file follow:
- ❒ \RS232\1 [6 1 2 1]
- ❒ \RS232\3 [4 1 2 1]
- ❒ \RS232\4 [7 1 2 1 00]
- ❒ \RS232\6 [6 3 2 3]
- ❒ \RS232\7 [5 1 2 1]
- ❒ \RS232\8 [6 1 2 1 00]
- ❒ \RS232\9 [6 1 2 1 00]
- ❒ \IOTechIEEEConverter\1
- ❒ \Channel\1 [10]

❒ In the "Resources" section of the GRM#*.reg registry file create a "resource" entry for each instrument on the Node by copying the corresponding resource information for that instrument from the GRMBase.reg template registry file. Select a unit number for each instrument. Note that an entry with a bracketed section indicates that a complete computer system is being viewed as an instrument. Such a computer may have special resource information. Observe the trailing space inside of the closing bracket.

- ❒ Our WffTdf example line has a "TDF" computer being viewed as a resource. The bracketed information is the DNS-style name of the machine. The trailing space is so that the closing bracket will not be viewed as part of the DNS-style name. You could have optionally placed the numeric IP address (with a trailing space) in the brackets.

- ❒ Example lines for the "Resources" section of the "Device" GRM registry file follow:
  - ❒ \Aydin329A\1
  - ❒ \Decom7715\1
  - ❒ \GDP233\1
  - ❒ \HP3325B\1

- ❐ \Krohnhite3905B\1
- ❐ \MetrumBVLDS\1
- ❐ \WffTdf\1 [WffTdf]
- ❐ Count the number of entries in the "Resource" section. It should be the same as the number of entries in the "Port" section of your GRM#*.reg registry file. Take a minute to review the concepts behind a "connection" in a registry file where a "port" is bound to a "resource."
- ❐ In the "Connections" section of the GRM#*.reg registry file create a "connection" entry for each instrument "port / resource" pair on the Node by copying the corresponding connection information from the GRMBase.reg template registry file.
  - ❐ Example lines for the "Connections" section in the "Device" GRM registry file follow:
    - ❐ \RS232\3\MetrumBVLDS\1
    - ❐ \RS232\4\Decom7715\1
    - ❐ \RS232\6\Aydin329A\1
    - ❐ \RS232\7\HP3325B\1
    - ❐ \RS232\8\GDP233\1
    - ❐ \RS232\9\WffTdf\1
    - ❐ \RS232\1\IOTechIEEEConverter\1\Channel\1\Krohnhite3905B\1
  - ❐ Observe, for example, the "Connections" entry for the "Aydin329A" instrument. You will note that it is a combination of the entry from the "Port" section for a port of type "RS232", located on "COM 6" (with options [6 3 2 3]) and the entry from the "Resources" section for a resource of type "Aydin329A", unit #1. A "Connections" section entry will not have the port options, just the port type and the COM number in the "port" section. The same "Connections" entry will have both the resource type and the unit number in the "resource"
- ❐ Take a minute to review how GRM numbers are established for a Node networked computer.
- ❐ In the "Configuration" section of GRM#*.reg registry file set the value of the "ProgramId" to the position that this GRM resides in the GRM chain and ensure that the "MaxResources" count is sufficient for the number of resources declared in this registry. Remember that this count will factor into the number of resources in the "Master" GRM registry file. You may need to revisit the "Master" GRM registry file to tune its "MaxResources" value.

To Configure the (Node) Operations Registry File

- ❐ There are steps in this sequence that are similar to the "Device" registry file configuration.
- ❐ As you did for the Device registry file, open the template file "GRMBase.reg" from the c:\Node\Regs directory in Notepad.
- ❐ Do a "Save As" and save the template file into c:\Node\Regs directory using a name of GRM#*.reg where the "#*" is the number of the combination of the "Operations" GRM position in a list of chained GRMs and the number of the Node networked computer where this file resides.
  - ❐ Once again, the number of your Node may be determined by establishing the position of this Node within the collection of Node networked computers within a tracking station. Most likely it will be the same as where the "Device" GRM resides that you just completed. You will then determine where this registry file lives along a sequence of chained together General Resource Managers. If there is only one Node in a tracking station, then the position of this Node is obviously 1. If there are more Node networked computers, then they will be numbered sequentially starting with 1. The number of an "Operations" GRM in a multi-node tracking station is "*#", where the "*" is the number of the Node in the list of Node networked computers in a tracking station and the "#" is the position of the "Operations" GRM within the collection of chained together GRMs. If there is only one Node in a tracking station then the first digit of "1" is omitted from the GRM number and is implied.
    - ❐ Example: The number of a "Operations" GRM in a single Node tracking station is "2", since it is the 2nd member of the GRM chain.
    - ❐ Example: The "Operations" GRM registry file on the 3rd Node networked computer in a tracking station would be named "GRM32.reg."
- ❐ Start a second copy of Notepad as you did with the Device registry file.

❐ Open the "GRMBase.reg" template registry file once again, from the c:\Node\Regs directory into the second copy of Notepad.
❐ Make a list of all types of instruments that will be controlled through the "Operations" GRM.
❐ In the "Port" section of your newly created "GRM#*.reg" Operations registry file create a "pipe port" entry for each instrument that you created a port entry for in the "Device" GRM registry file. Create the "pipe port" entries by copying the corresponding port information for that instrument from the GRMBase.reg file. The pipe port will be used by the device's corresponding operations. Also create a pipe port to be used to chain to and communicate with the "Device" GRM. Number the pipe ports sequentially starting with zero.
   ❐ Example lines for the "Port" section of the "Operations" GRM registry file follow:
      ❐ \Pipe\0
      ❐ \Pipe\1
      ❐ \Pipe\2
      ❐ \Pipe\3
      ❐ \Pipe\4
      ❐ \Pipe\5
      ❐ \Pipe\6
❐ Next chain the "Device" GRM to this "Operations" GRM. Chaining is done by creating a special "resource" for the "Device" GRM that is to be controlled by the "Operations" GRM in the "Resources" section of the "Operations" GRM#*.reg registry file Make sure that this GRM resource is listed before all of the other "instrument" resources.
❐ Continuing along in the "Resources" section create a "resource" entry for each type of instrument to be controlled in the "Device" GRM. Create the entries by copying the corresponding resource information for that instrument from the GRMBase.reg template registry file.
   ❐ Our Aydin329AOps example line has the "Operations Class" for an "Aydin329A" type of instrument. The bracketed information is the "ProgramId" of the "Operations" GRM which will communicate with the "Device" GRM along with the DNS-style name of the machine. You can optionally substitute the numeric IP address for the DNS-style machine name in the brackets.
   ❐ Example lines for the "Resources" section of the "Operations" GRM registry file follow:
      ❐ \GRM\3 [0 SGSNode]
      ❐ \MetrumBVLDSOps\1 [2 SGSNode]
      ❐ \Krohnhite3905BOps\1 [2 SGSNode]
      ❐ \HP3325BOps\1 [2 SGSNode]
      ❐ \Decom7715Ops\1 [2 SGSNode]
      ❐ \Aydin329AOps\1 [2 SGSNode]
      ❐ \GDP233Ops\1 [2 SGSNode]
❐ Count the number of entries in the "Resource" section. It should be the same as the number of entries in the "Port" section of your GRM#*.reg registry file. Take a minute to review the concepts behind a "connection" in a registry file where a "port" is bound to a "resource."
❐ In the "Connections" section of the GRM#*.reg registry file create a "connection" entry for each instrument type "port / resource" pair on the Node by copying the corresponding connection information from the GRMBase.reg template registry file.
   ❐ Example lines for the "Connections" section in the "Operations" GRM registry file follow:
      ❐ \Pipe\0\GRM\3
      ❐ \Pipe\1\MetrumBVLDSOps\1
      ❐ \Pipe\2\Krohnhite3905BOps\1
      ❐ \Pipe\3\HP3325BOps\1
      ❐ \Pipe\4\Decom7715Ops\1
      ❐ \Pipe\5\Aydin329AOps\1
      ❐ \Pipe\6\GDP233Ops\1
   ❐ Observe, for example, the "Connections" entry for the "Aydin329A" instrument. You will note that it is a combination of the entry from the "Port" section for a pipe port numbered "5", and the entry from the "Resources" section for a resource of type "Aydin329AOps", unit #1. A "Connections" section entry will not have the port options, just the number of the pipe port as designated in the "port" section. The same "Connections" entry will have both the resource "operations" type and the unit number.

❒   Note that in contrast to the "Device" GRM registry file, the ordering of the pipe ports, with the exception of the one selected for GRM use, is not critical. Recall that you must be mindful of the physical connection of an instrument to a specific COM port in the "Device" GRM registry, i.e. there must be a physical match of an instrument on the port and what is listed in the "Device" GRM registry. COM ports could be numbered anything, but must physically exist and have an actual instrument attached.

❒   On the other hand, in the "Operations" GRM registry it is different in that only the total number of pipe ports is of interest and that pipe ports must be sequentially numbered. Excepting the GRM pipe port, any "operations" instrument can be assigned to any pipe port within the list.

❒   Take a minute to review how GRM numbers are established for a Node networked computer.

❒   In the "Configuration" section of GRM#*.reg registry file set the value of the "ProgramId" to the position that this GRM resides in the GRM chain and ensure that the "MaxResources" count is sufficient for the number of resources declared in this registry. Remember that this count will factor into the number of resources in the "Master" GRM registry file. You may need to revisit the "Master" GRM registry file to tune its "MaxResources" value.

❒   Finish off registry file editing by revisiting "Master" registry file. As a final step you will tune the "MaxResources" entry count by ensuring that it is sufficient for the number of resources in the "Master" GRM registry and the GRM registries on all of the "Node" network computers.

Appendix I: Master/Node System Administration

Master PC Administration

There are many types of ATS files that are collected on the Master PCs during execution of ATS. These include message logs, SCC schedule and ephemeris tables, pass result summaries, operational profiles and device operation status files. All are retained and archived for thirty days before their automatic deletion from the system. Removal from the system is final; no intervening Windows "trash can" is utilized. The ATS process *PassResultsCompiler.exe* accomplishes the automatic removal task when it executes following takedown of any support.

Files automatically archived and deleted when a 30-day age is reached (or exceeded) includes:

File   Definition
C:\Master\Schedule\Archive\Prf*\Prfw*   Pass result summary files.
C:\Master\Schedule\Archive*.txt   Schedule/Ephemeris files delivered to SCC at initialization;
                                            ASCII pass log files retrieved from SCC at takedown.
C:\Master\MasterLogs*.txt   Master *MessageLog* files.
C:\Master\Schedule\Archive*.scd   WOTRS schedule files.
C:\Master\PtpLogs*   PTP operation logs collected from node at takedown.
C:\Master\TdfLogs*   TDF operation logs collected from node at takedown.
C:\temp*.log   All operation logs, including MetrumBVLDS status.

ATS application software is available on CD. One copy is retained at each ground station. See Appendix J: ATS Software Installation Guide for information regarding ATS installation.

Node PC Administration

The Node PCs are equipped with a 4.3-gigabyte drive. These PCs are responsible for starting and stopping operational equipment. They also poll status from the equipment and collect it in numerous files, or operation logs. These files are not removed by any process related to ATS. Instead, an operator or system administrator must delete them in order to avoid excessive system disk usage. Removal from the Windows desktop trash can is also required.

Files should be allowed to accumulate for a minimum of 30 days before their removal. No backup of these files is required. A procedure should be adopted which respects the following issues:

(1) Monitor node disk space usage. The accumulation of these files at a busy ground station will consume disk space quickly. Operational personnel should monitor disk space occasionally and take removal action when a 75-80% number is exceeded.
(2) File archive age. Files should be retained for a minimum of 30 days. An age of sixty or ninety days, however, may allow a longer interval for trouble-shooting of ATS problems.
(3) Assign a local system administrator. An on-site system administrator avoids delays associated with network activity from a remote user. It also provides operations personnel more flexibility regarding the method their machines should be managed.

The following files should be removed according to the procedure adopted at the ground station:

File   Definition
C:\Node\OperationLogs*.log   All operation logs collected during real-time support.

ATS application software is available on CD. One copy is retained at each ground station. See Appendix J: ATS Software Installation Guide for information regarding ATS installation.

Appendix J: ATS Software Installation Guide

ATS installation follows one of two procedures:

(1) **ATS Setup for New Machines**. This procedure installs all ATS software and creates required folders.

Resources Needed

- the Automated Tracking Station software on a CD, on a networked laptop, or on some other network resource.

Set the Source Directory

❒ Logon to the workstation as **Developer**.

❒ If installing from a networked laptop or other network resource, then map a network drive to the drive containing the ATS software.
  ❒ From the **Tools** menu of **Explorer** choose **Map Network Drive**.
  ❒ In the **Path** field type the name of the computer followed by the share name, i.e. \\ComputerName\ShareName. For example: \\LaptopName\C$ or \\WFF-Vandal\WOTS
  ❒ For **Connect As** type a valid logon name for the resource you are connecting to.

❒ Set the source directory for the installation files.
  ❒ From the **Control Panel** select **System**.
  ❒ From the **System** dialog select the **Environment** tab.
    ❒ In the **Variable** field type **srcpath**.
    ❒ In the **Value** field type the path to the directory of the ATS software. This may be the drive letter that was mapped above or the letter of the CD drive or a network share, followed by a directory name. The directory name will be the path required to reach the ATS installation files. This directory will contain the subdirectories Master, Node, General, and Install, among others. For example: e:\swInstall-07021999-ver23 for ATS release 2.3 installable from a distributed CD.
    ❒ Click **Set**, **Apply** then click **OK**.

Install the ATS Master Software

❒ At the workstation go to a DOS window command prompt.
❒ Change to the Install directory on the drive mapped above; for instance, : e:\swInstall-07021999-ver23 .
❒ Run the command line: **Install Master ATS Backup Update** replacing **ATS** with the designator of the station (AWOTS, MGS, AGS or SGS) being installed.
  ❒ Note: the command line is case-sensitive.
  ❒ Note: for a list of the command line arguments and available options, type **Install** with no arguments.
❒ Note: check the files in the Master\Station directory. If the ReadOnly attribute is present, remove it on each of the affected files.

Install the ATS Node Software

❒ At the workstation go to a command prompt.
❒ Change to the Install directory on the drive mapped above.

❒ Run the command line: **Install Node ATS Backup Update** replacing **ATS** with the designator of the station being installed.
  ❒ Note: the command line is case-sensitive.
  ❒ Note: for a list of the command line arguments and available options, type **Install** with no arguments.

(2) **ATS Updates for Previously-Installed Automation**. This procedure will install all executables, dynamic link libraries and text files. The zip file was created by ATS s/w personnel and can be installed at the operations personnel convenience.

Steps needed to install an ATS software upgrade that arrives in the format of a self-extracting ZIP file
❑ Start **Windows Explorer** and determine how your ATS upgrade file(s) were received
❑ If they came on a CD,
  ❑ Insert the installation CD into the drive and point Windows Explorer to the CD drive (generally D:).
❑ Else
  ❑ Point Windows Explorer to where E-mail attachments are stored or wherever you have this file locally saved (generally, c:\ATS-Zip).
❑ Determine if you will be installing an upgrade for a **MASTER** computer, or for a **NODE** computer, or for a combined **MASTER/NODE** computer.
❑ If combined **MASTER/NODE**
  ❑ Copy the Master, WFF and Node folders to a safe locations in case the new installation fails. Right click on the file "**swInstallMMDDYYYYver##.exe**" (where MM is the Month, DD is the Day and YYYY is the Year) and copy it to your "C:\temp" directory
❑ If MASTER only
  ❑ Copy the Master and WFF folders to a safe location in case the new installation fails. Right click on the file "**swMasterInstallMMDDYYYYver##.exe**" (where MM is the Month, DD is the Day, and YYYY is the Year) and copy it to your "C:\temp" directory
❑ If NODE only
  ❑ Copy the Node and WFF folders to a safe location in case the new installation fails. Right click on the file "**swNodeInstallMMDDYYYYver##.exe**" (where MM is the Month, DD is the Day, and YYYY is the Year) and copy it to your "C:\temp" directory
❑ Note: copying the zipped file to the "C:\temp" directory preserves the original file should you experience problems during the extraction and need to start the process over.
❑ Point Windows Explorer to your "C:\temp" directory and execute your selected exe file.
❑ Disregard the nag notice once the WinZip self-extraction engine starts
❑ When prompted for "**Unzip to folder**" select "**C:**" and not "C:\Temp" or any other directory on the fixed disk. You must use "**C:**" since there is directory information embedded in the zipped file that will be used to automatically direct where sub-directory files should go
❑ Check the "**Overwrite Files Without Prompting**" box if you don't wish to see what files are new since the last ATS software release.
❑ Select the "**Unzip**" button to start the unzip process. You do not need to start the WinZip program to complete file installation
❑ Shut down the self-extraction engine with the "**Close**" button. Your new files should now be in place and ready for use.

You may want to install a copy of **WinZip version 7.0 SR-1** on the target machine. Although **WinZip 7.0** is not necessary to run the self-extracting upgrade file, having it available will be especially handy in the future.

Steps to install the copy of **WinZip 7.0**
❑ Start **Windows Explorer**
❑ Determine how your WinZip upgrade file was received
❑ If it came on a CD,

- ❑ Insert the installation CD into the drive and point Windows Explorer to the CD drive (generally D:), but not always
- ❑ Else
  - ❑ Point Windows Explorer to where E-mail attachments are stored or wherever you have the needed file saved
- ❑ Right click on the file "**winzip70.exe**" and copy it to your "**C:\temp**" directory
- ❑ Point Windows Explorer to your "**C:\temp**" directory and execute "**winzip70.exe**"
- ❑ Follow the prompts through the installation sequence. Accept the defaults.
- ❑ When asked for "**Wizard**" based interface or "**Classic**" based interface select "**Classic**"
- ❑ When offered in the "**Other Options**" screen, select "**Create Icon For WinZip Self-Extractor**"
- ❑ At the "**WinZip Tip-Of-The-Day**" choose your preference
- ❑ WinZip version 7.0 will auto-start and is now ready for use.

Appendix K: Duplicate Tape Report Format

```
Master Tape
  Source          Primary          Duplicate
                  Media ID         Media ID
 Field    1    5              15              25
          SIDRRTTTTTTTTTTDDDDDDDDDD
```

SID = 3-character ground station ID (WPS, SGS, AGS, MGS); ***automatically input***.
RR = 2-character recorder type (SS for Sony, AA for Ampex); ***automatically input***.

TTTTTTTTTT = 10-character master tape media identifier; ***multiple list box selection***.

DDDDDDDDDD = 10-character duplicate tape identifier; ***operator input required***.